

Table of Contents

[Overview](#)
[License and Warranty](#)
[Installation and Removal](#)

[Hints and Tips](#)
[New for This Release](#)
[Stiletto and Windows 95](#)

[New User Information](#)

Functions:

[Configuring Stiletto](#)
[Info](#)
[Buttons](#)
[Menu Setup](#)
[Menu Contents](#)
[Alarm Setup](#)
[Alarm Details](#)
[Sound](#)
[Wallpaper](#)
[Hot Keys](#)

[Using Stiletto as Your Shell](#)

How To:

[Setup Stiletto](#)

[Access the Configure Stiletto Command](#)
[Active Window Switch with a Button Bar](#)
[Add a Reminder Message Alarm](#)
[Change Alarm Sounds](#)
[Change Button or Bar Size](#)
[Change Button Commands, Format, Position, or Number of Buttons Displayed](#)
[Change the Launch Menu Display](#)

[Change the Mouse Click Used to Display the Launch Menu](#)
[Change the Launch Menu Contents](#)
[Change System and Application Sounds](#)
[Change Wallpaper](#)
[Close an Active Window](#)

[Continue to Access the Desktop Menu from Your Program Shell](#)
[Cut and Paste Program Manager Commands to Stiletto](#)

Display Button Commands with Flyover Help

Drag and Drop to Start a Command

Drag and Drop a File Rename

Enter Commands

Move Stiletto

Put an Active Window Always On Top

Position Stiletto in the Active Window Caption

Put Stiletto always on top of the desktop

Restore the Default Stiletto Configuration

Run Commands from Program Manager or Other Shell Groups

Run Many Stiletto Button Bars

Run a Command at a Specified Time

Re-order buttons

Set a Low Resource Warning

Send keys to another Program

Set Alarms

Simulate Middle Mouse Button

Switch to Another Active Window

Start a Command as Always On Top

View Commands for a Button

View Windows Information

Overview

Stiletto (v1.9u1) is intended for people who want a button bar task switcher/launcher and clock/calendar which always stays at the same place on the desktop and which lets them use any mouse button to run a command. **Stiletto** also includes alarms, task scheduling, sounds control, and wallpaper control: it handles all your desktop needs using a single consistent but unobtrusive interface.

Stiletto incorporates these features:

- A small-footprint button bar launcher (e.g. fits over title bar of maximized window).

- Use of any mouse button to launch commands.

- Drag and drop files to start commands.

- User-configurable command launch menus with submenus.

- Attach launch menu to button bar, desktop, title bar, or all three.

- Start commands from hot keys or screen corners.

- Multiple button bars (through multiple instances and configurations).

- Floating button bar, choice of many resolution-independent standard positions, or position in caption of active window.

- Direct access to program manager or other desktop shell groups and their commands.

- Switch to or close any active window using a menu or button bar.

- Make any active task or launch any command as always on top.

- Text label, icon, clock, date, stopwatch timer, or resource display on any button.

- Adjustable button size.

- Built-in commands for drag and drop file rename, windows exit/restart, browsing and running files (with run history), moving the button bar, playing sounds, and others.

- Control of Caps Lock/Shift and Scroll Lock behaviour.

- Support for file associations.

- Reminder messages, regular chimes, and scheduled start/stop of commands.

- Wallpaper display and switcher/randomizer.

- Assignment of sound to application events.

- Randomization and testing of system and application sounds.

- Random noise making by playing sounds sporadically as specified.

- Send key strokes to programs when launched, or to running programs.

- Can act as your day-to-day shell

- Flyover (balloon) help to display the commands for any button.

- Compatibility with Windows 95, including support for long file names.

Stiletto is **not** intended to completely replace Program Manager or another **Windows shell** that you are using, as defined on the shell= line of your System.ini file, because it does not have capabilities to add or change icons. Instead, **Stiletto** is intended to provide quick, minimal-mouse click access to your most used commands while taking up little desktop space, and to provide utilities related to **Windows** start-up and time, all in one consistent package.

Creating Menus from Files or PM Groups

Using the builtin Files/Groups Menu command, you can create a menu which includes the files from a directory tree or Program Manager groups and then select an entry from the menu to execute that file or program.

This command might be useful to access directories where you keep documents that you are working on but which you do not access often enough to put on a **Stiletto** menu or button.

Similarly, you can use the Groups form of the command to put all of your Program Manager groups on a menu for access to programs you do not run often enough to put on **Stiletto** buttons or menus.

If you are running Win95, you can also use the command to access the Win95 programs and recently executed files directories. (**Stiletto** will add executed documents to recent directory if a check box is set on the menu setup dialog).

The parameters entry for the command is used to indicate the directory/PM Groups containing the files to be displayed. For Win95, you can use the special names "Recent" for the Win95 recent documents list; "Start Menu" for the Start Menu, and "Programs" for Start Menu programs. Or, set the parameters edit box to "Groups" for all PM groups or to "Group xxx" for the PM group named xxx. (This only works if you are using Program Manager as your shell).

The work directory edit box is used to hold keywords which control which files are displayed and how they are displayed:

Placing **nosubdir** in the edit box means no subdirectories will be included.

Placing **nosubmenu** in the edit box means all files from subdirectories will be listed in the main menu.

To include files with only certain extensions, list the extensions separated by blanks including the initial period.

To exclude files with certain extensions, list the extensions to be excluded, separated by blanks, and include a - in front of the period of each extension.

Examples:

nosubdir .exe 15 Include .exe files accessed less than 15 days ago from
main directory

.xls nosubmenu Include Excel spreadsheets from all subdirectories on
one menu.

-.dll -.bak Exclude dll and bak files.

If the Document/File Menu command is embedded in menu, it will be replaced by the contents of the generated menu.

Stiletto Built-In Commands

Stiletto comes with a set of built-in commands.

You will find the built in commands in the drop down box of the **Stiletto command entry controls** on the button configuration dialog, launch menu dialog, hot keys and alarms dialog.

Starting with version 1.9t, built-in commands are prefixed with an asterisk (*) to avoid conflicts with long file names in Windows 95.

This drop down box also allows you to select one of the Windows applets, like File Manager or Control Panel.

Following is an explanation of each built-in command:

Menu Separator	Separates displayed menu entries; only available for <u>launch menus</u> .
Start Submenu	Starts a <u>submenu</u> ; only available for <u>launch menus</u> .
End Submenu	Ends a <u>submenu</u> ; only available for <u>launch menus</u> .
New Menu Column	Starts a new column in the menu; only available for <u>launch menus</u> .
Active windows list	Inserts a <u>list of active windows</u> into the launch menu. When a list item is selected, that window is activated. Only available from <u>launch menu</u> .
Close Windows List	Inserts a <u>list of active windows</u> into the menu. When a list item is selected, that window is ended. Only available from <u>launch menu</u> .
Window On Top List	Inserts <u>list of active windows</u> into the menu. When list item is selected, that window is set to always on top. Only available from <u>launch menu</u> .
Window Not On Top List	Inserts <u>list of active windows</u> into the menu. When list item is selected, that window will no longer be always on top. Only available from <u>launch menu</u> .
Minimize Window List	Inserts <u>list of active windows</u> into the menu. When list item is selected, that window will be minimized. Only available from <u>launch menu</u> .
Recently Run Cmds List	Inserts a list of the 10 most recently run command lines into the menu. When a list item is selected, that command is executed. If the last command was the name of a document, the command line will consist of the associated .exe file followed by the document name (only the .exe file in Win95). Only

available from launch menu. (In rare cases, this command may cause GPFs when used with File Manager; if so, you will need to remove it).

Message Box	Displays a message box containing the text typed into the work directory of the command entry controls. It is intended for alarms, but can be used anywhere.
Launch Menu	Displays the Stiletto <u>launch menu</u> whose name is entered in the command parameters field.
Move Stiletto	Makes the Stiletto button bar <u>move</u> to the opposite side of the desktop.
Position Stiletto	You can drag Stiletto to an arbitrary <u>position</u> by clicking and dragging on a button which has been assigned this command.
Configure Stiletto	Activates the Stiletto <u>Configuring Stiletto</u> tabbed folder. Normally, the Info tab is displayed first, but you can select another tab by entering its index number (1 to 9) in the command parameters field. You can also start the Configuration dialog by left-clicking anywhere on the Stiletto bar with the Ctrl key pressed.
Close/... Window Under Mouse	Close/min/max/hide/on top/not on top the window underneath the mouse. These commands act on the window under the mouse cursor and so are most useful for Stiletto in to the <u>active window caption</u> or on a <u>menu.0</u> set to appear when the mouse is clicked on the a window or window title. The action applies to where the mouse was originally clicked, not the menu selection position. If the mouse is clicked on an MDI child window, the action applies only to that window; else it applies to the main, top-level window.
Exit Stiletto	Terminates Stiletto .
Browse and Run	Displays a file selection dialog; the selected file is executed. Stiletto will follow any File Associations when executing the selected file. Optionally, you can put a default starting directory in the work directory edit box. The browse dialog contains a history of commands/parameters and an edit box for new command parameters. You can also specify the default file extensions; see <u>Browse and Run</u> for details.
Type and Run	Displays a dialog with an combo box in which you can type the name of a file (with parameters) or document to run. Stiletto will follow any File Associations when executing the entered file. The combo box contains a history of commands/parameters from which you can select the file to be run.
Minimize All	Minimizes all top-level windows.

Bring Stiletto to Top	Brings every Stiletto button bar to top, making them visible (but does not change whether they are permanently on top). To bring only one button bar to the top, put its configuration file (ini file) name in the command parameters edit box
Hide Stiletto	Hides every Stiletto button bar, making them invisible. To hide only one button bar, put its configuration file (ini file) name in the command parameters edit box
Reset All Stiletto	Restarts all Stiletto bars (useful eg if you dynamically change the display configuration in Win 95)
Bar Size	The command Bar Size n changes the <u>number of buttons displayed</u> .
Prog Mgr/Shell Grps	Lets you execute a command from any existing <u>program manager or other desktop shell group</u> .
Hide Program Mgr	Hides Program Manager Window/Icon
Show Program Mgr	Shows Program Manager Window/Icon
Built-In Cmds List	Displays a list box from which you can execute any of the Stiletto built-in commands. To have the list sorted when the command starts, put the word "sort" in the command parameter field.
Hide Window	Used to select and then <u>hide a window</u> .
Reset Timer	Resets the <u>timer</u> associated with the button label.
Start Timer	Starts the <u>timer</u> associated with the button label.
Stop Timer	Stops the <u>timer</u> associated with the button label.
Clear Timer	Zeros the <u>timer</u> associated with the button label.
Toggle Timer	Stops the <u>timer</u> associated with the button label if it is running; starts it if it is stopped.
Drag and Drop Rename	Used to <u>rename</u> a file.
Quick Win Exit	Exits Windows to DOS (you will get a chance to save unsaved documents).
Quick Win Restart	Exits Windows to DOS and then restarts Windows (you will get a chance to save unsaved documents).
Quick System Boot	Exits Windows and re-boots DOS (you will get a chance to save

	unsaved documents).
Confirmed Win Exit	After asking you to confirm, exits Windows to DOS (you will get a chance to save unsaved documents).
Confirmed Win Restart	After asking you to confirm, exits Windows to DOS and then restarts Windows (you will get a chance to save unsaved documents).
Confirmed System Boot	After asking you to confirm, exits Windows and re-boots DOS (you will get a chance to save unsaved documents).
Windows Shutdown	Activates standard Windows shutdown dialog.
Execute Dos Command	Exits Windows, runs Dos command that is specified in command parameters field, and then restarts Windows. Make sure you specify entire path for Dos command. Only valid for win16 version of Stiletto .
Restart Stiletto	Restarts Stiletto based on the disk image of the ini file; to use a new ini file put its name, including the file extension, in the command parameters (it must be in the same directory as the current ini file).
Change Paper	Changes the wallpaper using the method selected in the <u>Wallpaper dialog</u> . If no change is selected, the wallpaper is changed to a random .bmp file in the same directory as the current wallpaper file.
Change Sound	Changes the sound files associated with all sounds in the <u>Sound dialog</u> that are not set to the no change sound method. Note that this is different from the Change Wallpaper command: the Change Wallpaper command always changes the wallpaper but the Change Sound command only changes sounds where the method for changing is not set to no change.
Play Sound	Plays a wav file drag/dropped onto the button. Stops any playing sound if no file is drag/dropped. You can also associate Stiletto with the WAV extension using File Manager. Double clicking on a WAV file in File Manager will then play that sound file.
Show Wallpaper	Shows wallpaper drag/dropped onto the button. Does not save it (use <u>Wallpaper</u> dialog for this). If no file dropped, shows wallpaper saved in configuration file. You can also associate Stiletto with the BMP extension using File Manager. Double clicking on a BMP file in File Manager will then preview that file as wallpaper.
Add Reminder Msg	Add a reminder message <u>alarm</u> .
Start Screen Saver	Starts your screen saver.

Enable Screen Saver	Enables your screen saver.
Disable Screen Saver	Disables your screen saver.
Send Keys	Used for <u>sending keys</u> .
Auto Minimize Mode	Turns off or on <u>autominimize</u> .
Stiletto Exec	Used for <u>close/min</u> and <u>filing</u> commands.
Files/Groups Menu	Create a menu of <u>directories or Program Manager groups</u> .
Prog Mgr Files	Lets you execute a command from any existing <u>program manager group</u> . Windows program manager need not be running; this command works directly from the program manager files. Only valid for win16 version of Stiletto .

Command Entry Controls

Stiletto uses the same set of controls to enter commands for button configuration, launch menu, hot keys and alarms.

With these controls you enter the file to be run when you press a button, or select a menu item, or when an alarm rings. You can enter a program file (.exe file) or a file name with an associated command (eg .xls file for Microsoft Excel). **Stiletto** provides many ways to find the file name: you do not need to type it in. You can browse for it, drag and drop it from file manager, cut and paste it from a program manager command, or capture it from a currently running program. You can also select a built-in command.

You also use the controls to set the start up (working) directory for the command and the initial window size for the command.

You can indicate that **Stiletto** should always launch a new instance of a program, or should switch to an already running instance if it exists. (Note, however, that many programs will only allow one instance of themselves to be launched).

Click on the following graphic of the command entry controls to learn about them.

Switch to if already active

To enter the command you want to run:

If you know the file name (ie the .exe file for commands or your file name for documents with associated commands:)

- type it directly into the edit box,
- or Browse for the command using the Browse Button,
- or drag and drop it from file manager onto the command edit combo box.

If you want to use a built in command or a Windows applet like File Manager:

- select one of the built-in commands from the drop down box,
- or right-click on the edit box and select a command from the alphabetical menu.

If you do not know the file name (.exe file) for your command:

- cut and paste it from a program manager group,
- or start the command you want to use and then use the capture button.

Stiletto supports file associations, so you can enter the name of a document file and **Stiletto** will run the associated command. For example, if you are working on a **Microsoft Excel** spreadsheet called mysheet.xls, you can insert this file name as a command (or browse for it and select it). **Stiletto** will run **Excel** on mysheet.xls when you select the command or when the alarm corresponding to the command rings

You can put command parameters in the parameters edit box. Or, to prompt for command parameters, put a question mark (?) in the command parameter box. You can put a ? in the midst of other command parameter text, and **Stiletto** will replace the ? with the prompted-for text.

To determine how the window will look when the command starts: Using the drop down box under the command entry combo box, you can determine whether the command starts in an ordinary window, minimized, or maximized. You can also specify that the command

windows should start as Always On Top. Finally, you can specify that the window from the command be hidden: this is intended for commands started by alarms.

To set the work (start-up) directory for a command: type the directory name into the work directory edit box or press the button underneath this edit box to set the work directory to the same directory as the command directory.

You also use the work directory edit box to:

Enter a message for the Message Box command built-in.

Enter a default group for the Program Mgr Groups built-in.

Enter a default starting directory and file extensions for the Browse and Run built-in.

Enter *close or *wait for close and wait alarms.

Enter |filepath.wav to play the sound filepath.wav when the command is started. If you also want to use a work directory, put the |filepath.wav immediately after the work directory name. (Works for all commands, including built-ins, except for above four).

Switching to another Active Window

Stiletto provides the capability to switch among active top-level windows using either a menu or a button bar.

To use the menu, you need to put the built-in command Active Window List directly into the launch menu or as part of a submenu of the launch menu. When the launch menu is activated, a list of currently active windows is displayed. Selecting one of the windows causes that window to be activated. It will be opened and put on top of the desktop if it is minimized or hidden beneath other windows.

Use the Menu Setup dialog to control whether or not hidden windows appear on the active window list, what prefix text to put in front of each entry of the menu to remind you that the selected entry will be activated, and whether **Stiletto** centers the window when it is activated.

You can ask **Stiletto** to automatically minimize non-active windows

Submenus

The Menu Contents dialog allows submenus to be created. When a launch menu is activated, the titles for these submenus are shown in the main menu. If you select one of these submenu titles, the submenu will be displayed and you can then select one of its entries.

The entries for submenus are shown indented in a launch menu list. The start of the submenu is marked by >> in the left margin. The end of the submenu is marked by << in the left margin.

To create a new submenu:

Select the Begin Submenu built-in in command.

Set the menu name to the submenu title.

Insert the Begin Submenu into the launch menu list box with the Replace or Add buttons.

Insert all commands for the submenu.

Select the End Submenu command.

Insert the End Submenu command

To change an existing submenu: delete, cut, copy, add, replace any of its entries.

To remove a submenu: delete the Begin Submenu and End Submenu commands.

To move existing commands onto a submenu: add a Begin Submenu command before the existing commands and add an End Submenu command after the existing commands.

Closing an Active Window

Stiletto provides the capability to close any active window using either a menu or a button bar.

To use the menu, you need to put the built-in command Close Window List directly into a main launch menu or as part of a sub-menu of a launch menu. When a launch menu is activated, a list of currently active windows is displayed. Selecting one of the windows causes that window to be closed. You will first be asked if you want to save any unsaved information.

Use the Menu Setup dialog to control whether or not Hidden windows appear on the close window active window list and to specify prefix text to put in front of each entry of the menu to remind you that you that the selected entry will be closed.

You can also close an active task with an alarm or a **Stiletto** command or close a **Stiletto** bar from a command

Making a Window Not Always On Top

Stiletto provides the capability to stop a window staying always on top of the other windows on your desktop.

To use this capability, you need to put the built-in command Window Not On Top List directly into the main launch menu or as part of a sub-menu of a launch menu. When a launch menu is activated, a list of currently active windows is displayed. Selecting one of the windows causes the window to be no longer be fixed on top of other task windows.

Use the Menu Setup dialog to control whether or not Hidden windows appear on the put on top active window list and to specify prefix text to put in front of each entry of the menu to remind you that you that the selected entry will be no longer be on top.

You can put windows on top with Windows on Top.

Making a Window "Always On Top"

Stiletto provides the capability to make any window stay always on top of the other windows on your desktop.

To use this capability, you need to put the built-in command Window On Top List directly into the main launch menu or as part of a sub-menu of a launch menu. When a launch menu is activated, a list of currently active windows is displayed. Selecting one of the windows causes the window to be fixed on top of other task windows.

Use the Menu Setup dialog to control whether or not Hidden windows appear on the put on top active window list and to specify prefix text to put in front of each entry of the menu to remind you that you that the selected entry will be put on top.

You can reverse the on top status with Windows Not On Top.

The Info Tab

The info dialog appears whenever you select the Configure Stiletto built-in command. It corresponds to the Info tab.

The left hand list box shows how long Windows has been running, what type of CPU you have, whether or not there is a math co-processor present, what mode Windows is running in, and the name of the configuration file used by this instance of **Stiletto**.

The right hand list box shows the percent of GDI and user resources free, the number of kilobytes of free memory, the number of tasks running, and the number of free kilobytes on each hard disk drive.

The Program Mgr/Shell Groups button starts the program manager/other shell dialog to help you cut and paste commands to **Stiletto**.

The Change Shell button lets you change the shell.

Use the alarm setup dialog to set a low resource warning alarm.

Configuring a Button

You use the Configure Button dialog to set the label of a button and the commands which are run when you click the button with the left, middle, or right mouse button.

(You can simulate the middle mouse button by holding down shift while pressing the left button).

This dialog can be accessed by pressing and holding the button to be configured, by selecting the button to be configured from the Configure All Buttons dialog, or by dropping a file on the button with the **Alt**-key held down, or under Windows 95, by right-dropping a file to a button.

To set the button label, you can:

- type a label of up to 31 characters into the label combo box .
- select a time or date display from the label drop down lists.
- indicate that an icon should be displayed.
- select one of the resource usage displays from the drop down box.
- select a timer as a label; if you do so, you can also assign a timer id.

To set the command to be run for a mouse button, use the command entry controls.

You can optionally enter help text to be displayed with flyover help.

You can change the size of an individual button by setting the Width Magnify percentage (for horizontal button bars) or Height Magnify percentage (for vertical button bars).

You can set this button's face and text colours by checking the colour checkbox and selecting the colour after pressing the text and face buttons. If you do not set both face and text, defaults are used (grey for face, black for text). Unchecking the box causes the button to revert to the bar colours.

The Configure All Buttons dialog lets you set the bar and base button size, bar orientation, button and flyover colour, font for labels, and number of buttons.

If you have swapped the mouse buttons using Control Panel - Mouse, then **Stiletto** will try to automatically swap the right and left labels on the button dialog. Automatic label swapping does not work with all mouse drivers, so you can also do it manually with the Configure All Buttons dialog.

If you drag/drop a file on a button with the **Alt** key held down (or right-drop under Win95), the configure dialog is started and the dropped file is used to initialize the command entry controls. If **Shift** is also held down, the middle button is initialized; if **Ctrl** is held down, the right button is initialized; otherwise, the left button is initialized.

(When you use a time, date, or resource as the button label, it is harder to remember what the button does. If you are not using flyover help, you can press and hold the button to view the function of a button if you forget. You can also assign common, easily-remembered functions to time/date buttons. For example, **Stiletto** comes with the time-labeled button configured to Configure Stiletto (left button) and to activate the built-in Move command (right button); the date-labeled button activates Windows Calendar (left mouse) and Windows Calculator (right mouse)).

The Buttons Dialog

The Button dialog is selected by clicking on the Buttons tab from the Configuring Stiletto command.

You can use this dialog to:

Set the position for **Stiletto** on your desktop (see below).
Configure any button through the Configure Button dialog.
Indicate whether **Stiletto** is to float always on top of all other windows.

Indicate whether all buttons are to be big (see below). This setting is also useful if you want to individually tailor the button sizes (using the Configure Button dialog) since it causes all buttons to start at the same size
Control if **Stiletto** shows a flyover help window for button commands.
Control whether a larger or small font is used for flyover help.
Control whether flyover help is shown in upper case, for greater legibility.

Determine the delay between the time the cursor is placed over the button bar and the time flyover help appears (drag/drop flyover help appears twice as fast).
Indicate that mouse button labels are to be swapped in flyover help.
Set the background and text colours for flyover help.

Determine whether **Stiletto** uses the **Windows** button colors or a button colour you choose with the Button colours dialog. **Windows** button colours are set in the Control Panel, Color, Color Palette dialog. Uncheck this box if you want to use a custom button colour.

Determine whether **Stiletto** automatically sets the number of displayed buttons to the highest button which is used (ie which has an associated command).

Determine whether **Stiletto** is hidden (made invisible) after each command launched from **Stiletto**. Before using this option, make sure you have a way to make **Stiletto** visible again! You could use the mouse cursor option (see next paragraph) or attach the built-in command Bring **Stiletto** to Top to launch menu Q or to another Stiletto button bar which is running. If you forget to do this, edit the ini file and remove the autohide option from the [General] section.

Determine whether **Stiletto** should be made visible when the mouse cursor passes over the nearest screen edge. **Stiletto** will be shown if invisible or uncovered if hidden by another window. For vertical **Stiletto**, the nearer vertical screen edge is used; for a horizontal button bar, the nearer horizontal screen edge is used. For this option, if AutoHide is checked, **Stiletto** will become invisible again when you move the mouse cursor off the button bar, even if no command is executed. You can control how long **Stiletto** stays visible with the autohide delay drop down. Use Bar Size 0 to make **Stiletto** invisible at start up.

Set the number of buttons displayed. (See also the size command).
Set the font to be used for the label of big or small buttons.
Set buttons and text colour (**Windows** button colours checkbox must be clear).
Determine if button text labels should be vertical or rotated: you will have to experiment with these settings, the text font, and the button sizes to get the best

effect. Not all fonts can be rotated; in particular, "small fonts" cannot. Note: starting with version 1.9t, to access rotated text you have to use an internal option.

Change the order of buttons.

Change the base button size.

You can specify that all buttons on the **Stiletto** button bar are to be big. **Stiletto** will use longer buttons for time and date and shorter buttons for any other label. This is in line with the **Stiletto** goal of having a minimal desktop footprint. However, it does mean that the label for shorter buttons must be brief. If you prefer to have room for a longer label, check the All Buttons Big check box.

Click on a button in the following graphic from the Buttons dialog to find out more about **Stiletto** positions.



You can also position **Stiletto** manually using the **Stiletto** Position built-in command.

Menu Setup Tab

The menu setup Dialog is selected by clicking on the menu setup tab from the Configuring Stiletto dialog.

This dialog controls: what appears on menus listing active tasks, shell-related functions, menu appearance, and how launch menu 0 can be made to appear by clicking on the desk top or a window.

Use the dialog to set a prefix string and whether or not hidden windows are displayed for the active window list, close window list, put-on-top window list, and put-not on-top window list.

There are check boxes for controlling Caps Lock and Scroll Lock keys.

There is a check boxes and edit boxes to control Auto Minimize Mode and Auto Hide.

You can indicate that windows should be centered when switched-to from the active window list or the Active Bar.

If you are configuring an Active Bar there will be a checkbox allowing you to specify that only one window per active task be represented on the active bar. You can also specify that minimized (iconized) non-Dos windows be hidden and whether other hidden windows should be displayed on the Active Bar. Finally, you can specify whether the right or middle mouse click should be used to close tasks on the Active Bar.

To obtain a pristine desktop, you can use **Stiletto** as your shell, or you can specify that the Program Manager icon/window should be hidden. **Stiletto** will wait for five seconds after it starts to hide the icon. This option can be used with the built-in Hide Program Manager and Show Program Manager commands, but acts independently of them. You can change the default five second delay with an internal option.

You can also enter an omit list of comma-separated strings which will be deleted from the list of active windows and Active Bar or which can be used to remove active windows entirely from these two.

For Win3.1x, you can indicated that right-clicking on a specified part of a title bar closes the associated window (you will be prompted to save unsaved info). As long as both settings are made from the same instance of **Stiletto**, this setting takes precedence over the title bar setting for launch menu 0. Right-clicking an MDI window usually closes only that window, but put RightCloseMDI=0 in the [general] section of Stiletto.ini to close the parent window instead.

You can enter a list of comma-separated strings for window captions of icons which are never to be hidden by the hide iconized windows option. Use a star followed by a comma at the end of a string to match all window captions starting with that string.

You can specify that the commands on a launch menu should be run each time **Stiletto starts**.

You can change the menu look. You can set the font for menus (to return to the standard font, reset the font to "System"). You can also ask for menus to have an OS/2 look, Win 95 look, or button look. Non-standard menus will use the Windows button face colour (since

the default Windows menu colour is white which spoils the 3D effect) and the Windows menu highlight colour, unless you check the colour box, in which case **Stiletto** will attempt to set the menu to have the same colours as the associated **Stiletto** button bar. (The close/on top prefixes do not currently work for non-standard menus).

Menu Contents Tab

Stiletto allows you to create menus of commands. When you select one of the menu entries, **Stiletto** runs ("launches") the corresponding command; hence these menus are called "launch menus".

One possible use for menus is to use them to access commands or files you use less often than the commands you assign directly to a button. Using this approach, you assign a few, most-used commands directly to buttons and many less-used commands to launch menus. Of course, this is just one idea; there are many other approaches (eg some people use no button bar and put all commands on launch menus).

You create or change launch menus with the Menu Contents tab, which is displayed when you click on Menu Contents on the Configuring Stiletto dialog, or when you click on a button which is assigned a Launch Menu command with the Alt key held down.

You run commands from launch menus with the built-in Launch Menu commands. A **Stiletto** launch menu is displayed when you click a **Stiletto** button which has been assigned the Launch Menu built-in command with a command parameter set to the launch menu name. Another way to access launch menu 0 is to click on the title bar or desktop with the mouse button selected in the Menu Setup dialog.

At the left of the menu contents tab is a list box which displays the currently selected launch menu. Submenus are shown indented. You can select which menu is displayed with the drop down box labeled Menu Name. You can add or rename (up to 50) menus with the add or rename buttons. You can delete menus by deleting each individual menu entry.

In the middle of the launch menu dialog are a set of buttons used to transfer information between the launch menu list box and the controls used to work with a single menu entry. These controls are only enabled when their function can be performed; for example, the Add After control is only enabled when you have entered a valid Menu Name and Command and have selected an item on the list to be added after.

At the right of the dialog are controls for changing, adding, or deleting menu lines (items). The adjacent edit box is used to enter the name of the menu item. The command entry controls beneath the edit box are used to set the commands to be run when the menu item is selected.

To add a new command to the menu:

Select the menu you want to use or add a new one. You may want to avoid modifying the name of Menu 0 until you are familiar with **Stiletto** operation.

Enter the menu item name on the right hand controls. If you want an hot-key (mnemonic) letter for the menu item, put an ampersand (&) in front of the letter in the name.

Enter the command, parameters, and work directory in the right hand controls.

If you want to add the entry before or after an existing entry, or replace an entry, select it.

Push the button corresponding to how you want to add the new entry to the list.

If you added a new menu, to access it assign the Launch Menu command to a button and set the parameters edit box to the new menu's name.

To change an existing entry:

Select the entry from the list.
Press copy (or double click on the entry).
Change the menu name or command as desired.
Press the replace button.

To move or copy an existing entry:

Select the entry to be moved or copied.
Press cut for a move, copy for a copy (or double click for copy).
If you to to move/copy to a different launch menu, select that menu number.
If you want to move/copy the entry before/after an existing entry, select that entry.
Push the button corresponding to how you want to add the new entry to the list.

To delete an entry:

Select the entry to be deleted.
Press the Delete button.

See submenus to learn about working with submenus.

When you click on a button which is assigned a launch menu command, **Stiletto** will display the menu: You can then release the button and select the desired item or, alternatively, move the pointer to the desired item and release. If you prefer the second method, **Stiletto** has an internal option to disable the first method.

Alarm Setup

The Alarm Setup dialog is displayed when the alarm setup tab is clicked from the Configuring Stiletto dialog.

Using this dialog you can set check boxes to:

Have **Stiletto** ring alarms which occur when **Stiletto** is not active. Otherwise, missed alarms are not rung but are recycled or discarded according to the alarm setting. (However, alarms less than four minutes old are always rung).

Play the alarm sound when an alarm displays a message box.

Play the alarm sound when a command is run by an alarm.

Keep a timer log.

Keep an alarm log.

Specify that seconds should be shown on timer labels.

The Alarm Setup dialog also contains several drop down lists which you use to:

Set a chime at a regular time during the hour (eg every 15 minutes).

Play the alarm sound at when at regular intervals for any timer.

Save timers at regular intervals so that **Windows** crashes do not cause autosave timer information to be lost.

Set a resource warning level percentage to have **Stiletto** display a message box whenever GDI or USER resources fall below this level. You can also monitor resource usage with a button label set by the Configure Button dialog.

Set the interval in seconds at which **Stiletto** updates timers, checks the resources, and updates any resource usage button labels set by the Configure Button dialog.

Use the Sound dialog to set the sound associated with alarms and chiming.

Alarm Details

The Alarm Details dialog is displayed when you click on the Alarm Details tab of the [Configuring Stiletto](#) dialog.

Use this dialog to set alarm times and the command or messages to be activated when the alarm rings.

The left side of the dialog shows a list box containing all current alarms, sorted so that the earliest is at the top.

The right side of the dialog contains a set of controls for entering or changing a single alarm, including the alarm time and date, a drop down box to specify what **Stiletto** should do with the alarm after it rings, and a set of [command entry controls](#) to enter the command to be executed or message to be displayed when the alarm rings (use the Message Box [built-in](#) command to display a message for an alarm).

To enter an alarm date, enter the year, month, and day as one or two digits separated by a space. When you enter a valid date, the day of week and date will be displayed beside the date you enter. You can use the "=" button to reset to the current date, and the "+" and "-" buttons to change the year, month, or day. (These buttons will repeat if held down.)

To enter an alarm time, enter the hour and minute as one or two digits separated by a space. Use a 24 hour clock or put a (for AM) or p (for PM) after the time. When you have entered a valid time, it will be displayed in AM/PM format beside the time you enter. You can use the "=" button to reset to the current time, and the "+" and "-" buttons to change the hour or minute. (These buttons will repeat if held down.)

(Note that **Stiletto** only checks to see if an alarm should be rung once per minute. If you set an alarm for now, it will not ring until the next minute).

In the middle of the dialog are a set of buttons which are only enabled when the corresponding action can be performed. For example, the Replace button is only enabled when you have entered a valid alarm time and date and command and have selected the alarm to be replaced from the alarm list.

Alarms are usually used to start commands, but you can also use alarms to [close or wait for running tasks](#).

To quickly add a new message box (reminder) alarm, see [Adding a Reminder](#).

When a message box alarm rings, you can change the message text and re-schedule it, if you like.

To add a new alarm:

Enter the alarm time and date.

Select the fate of the alarm after it has rung.

Enter a valid command, using Message Box [built-in](#) command for an alarm message.

Press the add button.

To replace or change an existing alarm:

Select the alarm.

Press copy or double click the alarm.

Enter the alarm time and date.

Select the fate of the alarm after it has rung

Enter a valid command, using Message Box built-in command for an alarm message.

Make sure the alarm you want to change is selected.

Press the Replace button.

To delete an alarm:

Select the alarm from the list.

Press the delete button.

Stiletto Sounds

Stiletto sounds are set from the [Sound dialog](#).

You must have a sound card and the appropriate drivers or the PC speaker driver to hear sounds in Windows.

Stiletto supports these sounds:

Windows Start:	Plays once when Windows starts.
Windows Exit	Plays just before Windows shuts down.
Asterisk	Plays when an Asterisk message is displayed by Windows.
Default Beep	Plays when a Windows programs asks for a beep.
Critical Stop	Plays when an Critical message is displayed by Windows.
Question	Plays when an Question message is displayed by Windows.
Exclamation	Plays when an Exclamation message is displayed by Windows.
Stiletto Chime	Plays whenever Stiletto chimes (see Alarm Setup dialog)
Stiletto Alarm	Plays whenever Stiletto alarms (see Alarm Setup dialog). Use an asterisk to have the PC Speaker beep for alarms.
Stiletto Noise	Windows background noise: sound plays at random minutes according to the percentage set in the Sound dialog.

The following sounds play for Win3.1x and only if the Enable Application Sounds box is checked in the Sound dialog.

App Start	Played when an application starts.
App End	Played when an application ends.
Win Minimize	Played when a window is minimized.
Win Maximize	Played when a window is maximized.
Win Move	Played when a window is moved or re-sized.
Dialog OK	Played when a dialog box OK button is pressed (if the application is using the standard OK internal message number).
Dialog Cancel	Played when a dialog box Cancel button is pressed (if the application is using the standard Cancel internal message number).

Sound Tab

The Sound dialog is selected by clicking the Sound tab on the Configuring Stiletto dialog

The Sound dialog is used to change the sound file (wav file) associated with any of the **Stiletto** sounds.

The left side of the dialog shows a list of the standard **Stiletto** sounds and any sound files currently assigned to the sound.

The right side of the dialog contains controls for working with a single sound and its sound file.

To use a sound file, type its name directly into the sound file edit box, drag and drop it from the file manager, or use one of the browse buttons to display and select from a list of wav files. If you want to hear the sounds as you browse them, press the Browse and Test button; clicking on a wav file in the file browse dialog will play that sound file.

To play the sound file shown in the sound file name edit box, press the Test button.

To add spice to your multimedia life, **Stiletto** can automatically change the sound file you select to any other wav file in the same directory. Use the How To Change Sound drop down box to determine whether and how **Stiletto** changes the sound file. Use the Mins Between Changes drop down box to determine how often **Stiletto** changes the sound file (this drop down is disabled if you selected No Change from the How To Change Sound drop down). For Win95, **Stiletto** can only change sounds at Windows Start Up.

The buttons in the middle of the dialog box are used to work with sounds from the list on the left. A button is only enabled when its action is valid; for example, the Replace button is only enabled when there is sound information in the right hand edit box and there is a sound selected from the list to be replaced.

To add or replace sound file information for a particular sound event:

Enter or browse for the sound file name.

Enter the How to Change Sound and Mins Between Sound, as required.

Select the sound with the sound information you wish to change or add.

Press the replace button.

To work with existing sound file information for a particular sound event:

Select the sound with the sound information you wish to use.

Press copy (or double click).

Change the sound file name, How to Change Sound, or Mins Between Sound.

Select the sound with the information to be changed.

Press the replace button.

To stop the sound for a particular event:

Select the sound to be cleared.

Press Clear.

The Enable Application Sounds check box must be checked to have **Stiletto** play a sound for any of App Start, App End, Win Minimize, Win Maximize, Win Move, Dialog Ok, and Dialog Cancel.

The **Stiletto** noise sound is played at random in the percentage of minutes set by the % of mins to play noise edit box. Use this feature if you feel drowsy. The noise sound is changed each time it is played, unless the change method is set to No Change.

You may wish to group your wav files into directories (eg startup, shutdown, good, bad) and then use the **Stiletto** automatic sound file changer to sample the sounds from each directory.

The Change Sound [built-in](#) command changes any sounds which have not been marked as No Change. If you assign this command to a button or put it on a **Stiletto** launch menu, you can use it to have **Stiletto** immediately change sounds if a sound pattern that you dislike is selected.

Wallpaper Tab

The Wallpaper dialog is selected by clicking the Wallpaper tab on the [Configuring Stiletto](#) dialog.

The Wallpaper dialog is used to change the Windows wallpaper.

To specify a wallpaper file, type its name directly into the wallpaper file name edit box, drag and drop it from the file manager, or use one of the browse buttons to display and select from a list of bmp files. If you want to see the wallpaper as you browse the files, press the Browse and Display button; clicking on a file in the file browse dialog will display that wallpaper.

To add spice to your multimedia life, **Stiletto** can automatically change the wallpaper you select to any other bmp file in the same directory. Use the Method for Changing Wallpaper drop down box to determine whether and how **Stiletto** changes the wallpaper file. Use the Mins Between Automatic Changes drop down box to determine how often **Stiletto** changes the wallpaper file (this drop down is disabled if you selected No Change from the Method for Changing Wallpaper drop down).

Use the Tile Wallpaper check box to set whether Windows tiles or centers your wallpaper.

Stiletto has a [built-in](#) Show Wallpaper command for previewing wallpaper.

You can also associate **Stiletto** with the bmp extension using File Manager. Double clicking on a bmp file in File Manager will then preview that file as wallpaper.

Viewing A Button's Commands

To view the commands associated with a **Stiletto** button bar button, press and hold down the button for about one second. Or you can rely on flyover help to show the commands for a button.

The Configure Button dialog will be displayed which will show you the command associated with each mouse button.

To execute one of these commands, press the **Stiletto** button by clicking the appropriate mouse button; the dialog box is automatically closed when you do this.

You can also press Esc or the Cancel button to close the dialog box.

Of course, you can also use the dialog box to change the button label or any command after you have displayed it by holding down the **Stiletto** button bar button.

Configuring Stiletto

You configure **Stiletto** buttons, launch menu contents, sounds, wallpaper, hot keys, and alarms with the Configure Stiletto built-in command, which displays a set of tabbed dialogs.

You can also start this dialog by left-clicking anywhere on the **Stiletto** bar with the Ctrl key pressed.

Normally, the Info tab is displayed first, but there are built-in Configure commands which go immediately to any other tab.

Any of these Configure commands can be assigned to a **Stiletto** button or to a launch menu. In **Stiletto** as originally shipped, the standard Configure command appears on a launch menu and is also associated with left-clicking the leftmost button on the button bar.

Re-installing the Configure command shows what to do if delete this command and need to configure **Stiletto**,

The Configure Stiletto command displays a set of tabbed dialogs as follows:

Info shows Windows information and resource usage.

Buttons allows button commands, position, font, number to be set.

Menu/ Setup controls the format of the active window list, the mouse click used to start launch menu 0, and the format of menus.

Menu Contents allows you to change the contents of launch menus.

Alarm Setup controls low resource warnings and the sounds associated with alarms and chimes.

Alarm Details allows you to add or change alarms.

Sound controls the sounds played for windows and application events and allows you to specify how **Stiletto** should automatically change them.

Paper allows you to change the Windows wallpaper and to set up automatic re-wallpapering.

Key/Mouse allows you to assign commands to hot keys or to screen corners.

All the tabbed dialogs share one OK, Apply, and Cancel button: If you press OK, all changes made on all tabbed dialogs are saved and **Stiletto** restarts with the new configuration. If you press Apply, all current configuration changes are saved. If you press Cancel, all changes made on any tabbed dialog since the last Apply are discarded.

Note however that the Configure Button dialog has its own OK and Cancel; if you press OK on this dialog the changes are saved even if you press Cancel on a tabbed dialog.

Stiletto License and Lack of Warranty

The **Stiletto** program, DLL, Help File, and readme file are all Copyright 1996 by Bruce Switzer. All Rights Reserved.

THIS SOFTWARE IS DISTRIBUTED "AS IS," WITHOUT WARRANTY AS TO PERFORMANCE OF MERCHANTABILITY OR ANY OTHER WARRANTIES WHETHER EXPRESSED OR IMPLIED. BECAUSE OF THE VARIOUS HARDWARE AND SOFTWARE ENVIRONMENTS INTO WHICH THIS PROGRAM MAY BE PUT, NO WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE IS OFFERED. GOOD DATA PROCESSING PROCEDURE DICTATES THAT ANY PROGRAM BE THOROUGHLY TESTED WITH NON-CRITICAL DATA BEFORE RELYING ON IT. THE USER MUST ASSUME THE ENTIRE RISK OF USING THE PROGRAM.

You can contact the author with comments or questions at bswitze@io.org or bruce.switzer@canrem.com.

Stiletto is freeware, but it is not in the public domain. It may be distributed freely but may not be sold nor be used in any profit-oriented endeavor. In particular, Stiletto is intended for private use or for use in academic/charitable institutions only. Stiletto may not be used in a for-profit corporation or the government unless a license is obtained from the author. Unlimited corporate or government department use licenses are available from the author for a small fee.

All trademarks used in this Help File are the property of their respective owners and are used for explanatory purposes only.

Moving Stiletto

Although it is generally more convenient to have the **Stiletto** button bar at a fixed place on the desktop, there are times when you want to temporarily move **Stiletto** out of the way.

The built-in Move **Stiletto** command does this. When executed, this command causes the button bar to jump to the other side of your Windows desktop. If you execute it again, **Stiletto** jumps back to its previous position.

(Or you can also have a floating button bar through the Position **Stiletto** built-in command).

The Move command can be assigned to a button on the button bar or put in the Launch Menu.

The Move command is only temporary; the next time **Stiletto** starts up, it will be displayed in its standard position.

Move handles the split button bar as follows: if the move command is assigned to the left half, then the button bar will be fused on the right half of the desktop. If the move command is assigned to the right half of the button bar, then the button bar will be fused on the left half of the desktop. (Hence you may want to assign the move command to two different buttons.) When the move command is pressed from a fused button bar, it will be resplit. Although this may seem complicated, the idea is that a single mouse click near the area you want to clear will move **Stiletto** out of the way.

Installation

Automatic Installation

To install **Stiletto**, first UnZip the **Stiletto** zip file into its own directory. Then execute the ssetup.exe program that comes as part of the zip. This program copies the 3d Control library and stilethk.dll to your Windows directory (if you do not already have the latest version of them).

After you are satisfied that **Stiletto** is working, you can delete the .bak files.

To start **Stiletto**, execute stiletto.exe. **Stiletto** will use the default configuration stored in stiletto.ini. You may also wish to set up a **Stiletto** Active Bar to track and switch among active tasks.

If you find **Stiletto** useful, you will probably want to put in it your StartUp group so that **Stiletto** starts automatically with Windows.

Starting with version 1.9q4, **Stiletto** stores stilethk.dll in the appropriate Windows directory. If you get "undefined dynalink" errors, it means you have an older version of stilethk.dll interfering with proper operation. Check your old **Stiletto** directories and delete or rename any obsolete versions of this file.

Manual Installation

To manually install **Stiletto**, first UnZip the **Stiletto** zip file into its own directory. Then move ctl3dv2.dll and stilethk.dll to your Windows\system (preferred) **or** Windows directory. Make sure there is no copy of either of these files in your **Stiletto** directory. Finally, if you do not have a stiletto.ini file from a previous version of **Stiletto**, rename stiletto.def to stiletto.ini.

To start **Stiletto**, execute stiletto.exe. **Stiletto** will use the default configuration stored in stiletto.ini. You may also wish to set up a **Stiletto** Active Bar to track and switch among active tasks.

If you find **Stiletto** useful, you will probably want to put in it your StartUp group so that **Stiletto** starts automatically with Windows. Or you may want to use Stiletto as your shell.

Starting with version 1.9q, **Stiletto** stores stilethk.dll in the appropriate Windows directory. If you get "undefined dynalink" errors, it means you have an older version of stilethk.dll interfering with proper operation. Check your old **Stiletto** directories and delete or rename any obsolete versions of this file.

Removal

To de-install **Stiletto**, remove it from your StartUp group and erase the directory containing **Stiletto**. Erase stiletto.ini and stilethk.dll from your Windows or Window\system directory if there. The 3D control library is used by many programs so you should leave it in your Windows directory (the file name is ctl3dv2.dll).

Default Configuration

The default **Stiletto** configuration is stored in the stiletto.ini file that comes with **Stiletto**. There is a backup copy in stiletto.cfg in case you ever want to return to the default configuration (by copying stiletto.cfg over top of stiletto.ini in your **Stiletto** or **Windows** directory).

(See [installation](#) for more about installing and de-installing **Stiletto**).

The default **Stiletto** button bar has six buttons with these contents:

Butt on	Label	Left	Middle	Right
1	12-Hour Clock	Stiletto Configure	Stiletto Move	Stiletto Position
2	Date mmm-dd	Win Calendar	None	Win Calculator
3	DOS	dosprmt.pif	Confirmed Restart	Confirmed Win Exit
4	f.s.c	File Manager	Win Setup	Control Panel
5	Edit	NotePad	Drag/Drop Rename	SysEdit
6	Menu	Launch Menu 0	Task Manager	Prog Mgr/Shell Grps

The default launch menu starts with three submenus: switching between active windows, closing active windows, putting an active window on top. Next comes the [built-in Stiletto](#) commands Prog Mgr/Shell Groups, Configure Stiletto, Browse and Run, Change Wallpaper, Change Sounds, and Exit **Stiletto**.

[Flyover](#) help is turned on in the default configuration. If you find it distracting, you can turn off with the [Configure All Buttons](#) dialog.

Stiletto will be positioned horizontally in the top-left corner of the desktop.

Stiletto will be positioned horizontally in the top left of the desktop offset to the right by the width of a system menu box.

Stiletto will be positioned vertically in the top left corner of the desktop.

Stiletto will be positioned vertically in the top right corner of the desktop.

Stiletto will be positioned horizontally in the top right corner of the desktop.

Stiletto will be positioned horizontally in the top right of the desktop, offset to the left by the width of a minimize box plus a maximize box.

Stiletto will be positioned vertically in the middle of the left side of the desktop.

Stiletto will be positioned vertically in the middle right hand side of the desktop.

Stiletto will be positioned vertically in the bottom left corner of the desktop.

Stiletto will be positioned horizontally in the bottom left corner of the desktop.

Stiletto will be positioned horizontally in middle of the bottom of the desktop.

Stiletto will be positioned horizontally in the bottom right corner of the desktop.

Stiletto will be positioned vertically in the bottom right corner of the desktop.

Stiletto will be positioned horizontally at the top and in the middle of the desktop.

Stiletto will be positioned on the right of the active window caption.

Stiletto will be positioned on the left (middle for Windows 95) of the active window caption.

Stiletto will be split in half and placed horizontally on each side of the top of the desktop. The left half will be positioned horizontally in the top left of the desktop offset to the right by the width of a system menu box; the right half will be positioned horizontally in the top right of the desktop, offset to the left by the width of a minimize box plus a maximize box.

Re-installing the Configuration Command

You use the Configuring Stiletto built-in command to set button commands, launch menu contents, sounds, wallpaper, hot keys, and alarms. This command is associated with the menu or button just like any other Windows or other built-in command. And, like them, it may be deleted.

What happens if you delete the Configuring Stiletto command from all buttons and the menu? How can you configure **Stiletto** then?

You can simply left-click on the button bar with the Ctrl key pressed. Or, here is another approach:

Press and hold any **Stiletto** button for about one second to display the Configure Button dialog.

Under Left, activate the drop down box and selected Configure **Stiletto** from the list. Press OK to close the dialog.

Left click the button you selected from the button bar to activate the Configure **Stiletto** dialog.

Now you can reconfigure all the buttons and launch menu and place the Configure **Stiletto** command wherever you wish.

If you have assigned launch menus or the **Stiletto** Position command to the left and right mouse buttons of all visible buttons, you will have to press and hold the middle button. You can use shift-left to do this.

Note that if you have assigned the Position **Stiletto** command to the button, when you press and hold you have to be careful not to move the mouse at all.

Simulating Middle Mouse Button

If you hold a shift key down while you press the left mouse button, **Stiletto** will act as if the middle mouse button has been pressed. (This applies only to selecting commands from the **Stiletto** button bar.)

Drag and Drop onto the Stiletto Button Bar

You can drag and drop a set of one or more file names from the File Manager onto the **Stiletto** bar to start a command with the file names as the parameters.

Normally, the command corresponding to the left mouse button is started. But if you hold down the **Shift** key as you drop the file name, the command corresponding to the middle button is started. If you hold down the **Ctrl** key, the command corresponding to the right mouse button is started.

You do not have to memorize this rule: if you hold the file to be dropped over the button bar momentarily without dropping it, **Stiletto** will display a small flyover window showing the three commands associated with the button. **Shf** will appear beside the middle mouse button command; **Ctl** will appear beside the right mouse button command.

(As another memory aid, consider this: both Shift and Middle have i in them; both Right and Ctrl have r in them.)

Stiletto does not check on the **Shift** or **Ctrl** keys until after you drop the file. If you are dropping a file from a floppy disk, you need to hold the key down for a moment after the file is dropped since **Windows** does not tell **Stiletto** that the file is dropped until after it checks the floppy drive which takes a second or two.

Stiletto has a built-in drag and drop rename command, Show Wallpaper command for previewing wallpaper, and Play Sound command for playing wav files.

Sometimes you want to drag and drop files in the middle of the command line. To do so, put the character "|" at the point where you want the dropped files to be placed. The "|" will be replaced by the dropped files when the command is run and the text following the "|" will follow the dropped files. Do not forget a space after the "|", if needed.

Stiletto drag and drop also works with the **Norton Desktop for Windows** file manager and the **SideBar** file manager. However, flyover help for drag and drop does not work with the file managers for these two desktops.

You can drag and drop files to the Active Bar, and they will be passed to the executing program (if the program does not accept dropped files, you will hear an error beep).

Stiletto always attempts to start a new instance of a command when a file is dropped on a button.

Drag and Drop Rename

The built-in drag and drop rename command can be used to rename a file with a new name or extension. The renamed file stays in the same directory.

File wildcards are not allowed.

You can associate this command with a launch menu or the button bar, but if you want to use the drag and drop capability, the command must be assigned to a button.

Activating the command brings up a small dialog box where you can type the current file path (drive, directory, file name, extension), the new file name, and the new extension.

If you drag and drop a file name from file manager onto the button with the rename command, the current path edit box will be initialized to the dropped file name and the new file name and extension edit boxes will be initialized to the dropped file name and extension.

If you Press the Ext to .bak button, the new file extension is set to .bak and the new file name is set to whatever was present in the new file name edit box.

Displaying Button Commands

If you hold the mouse button over the **Stiletto** button bar without pressing a mouse button, **Stiletto** will display a small window showing the commands associated with each of the three mouse buttons for the underlying button. Each of the letters **L**, **M**, or **R** (Left, Middle, Right) appears beside the appropriate command.

After you press a button or move the mouse off the **Stiletto** bar, this window will disappear.

If you are using drag and drop, **Shf** appears beside the command associated with the middle mouse button, since you need to hold down the **Shift** key to access the middle button command for drag and drop. Similarly, **Ctl** appears beside the command associated with the right mouse button, which is accessed by holding the **Ctrl** key down as you drop the file.

(**Stiletto** drag and drop also works with the **Norton Desktop for Windows** file manager and the **SideBar** file manager. However, flyover help for drag and drop does not work with the file managers for these two desktops).

You can control whether **Stiletto** displays the help window with the Configure All Buttons dialog from the Configuring Stiletto built-in command.

Use the Configure Button dialog to set the text displayed with flyover help. If you do not specify any text, the commands themselves are displayed.

For clearest display, you need "Small Fonts" installed to use flyover help.

If you have swapped the mouse buttons using Control Panel - Mouse, then **Stiletto** will try to automatically swap the right and left labels on the button dialog. Automatic label swapping does not work with all mouse drivers, so you can also do it manually with the Configure All Buttons dialog.

After you launch a command, flyover help is normally suspended until you move the mouse off the **Stiletto** button bar. This is to prevent problems with full screen **Dos** commands: if flyover help is not suspended, flyover help will appear unnecessarily after you activate a full screen **Dos** command and minimize the **Dos** window. But, if you never use full screen **Dos** and want flyover help not to be suspended, you can use an internal **Stiletto** ini option to accomplish this.

Accessing Program Manager or Other Shell Groups

You use the built-in Prog Mgr/Shell Groups command to run commands from your existing program manager or other desktop shell groups. This command will work with any shell which is fully compatible with Program Manager; it has been tested successfully with **SideBar** and **Norton Desktop 3**. (See the end of this section for the Program Mgr Files command, which always uses your Program Manager files).

The Prog Mgr/Shell Groups command displays a dialog box with two list boxes. The left list box shows the names of all your groups. Clicking on any one of these will fill the right list box with the names of all the commands (ie icons) in that group. You can then execute any command in the group by either double clicking on it, or by selecting it with a single click and pressing the Execute button.

The command dialog is normally closed when you run a command; however, if you uncheck the Close When Command Run check box, the dialog will stay open after a command is run.

If you single click on any command name in the right list box, you will see the file paths for the command and its work directory displayed at the bottom of the dialog box.

You can set a Hot Key to immediately access this command.

The command can also be used to cut and paste commands directly to **Stiletto** command entry controls.

You can set up the command to immediately display the commands for any group when started by typing the name of the default group into the work directory field of the command entry controls. For example, if you had several groups you accessed often, you could set up several different commands, one for each group, and assign them to a launch menu with the menu name set to the group name. Then, when you selected that item from the menu, the Groups command would start with the right hand list box filled with the commands from the default group.

(The built-in command Program Mgr Files is used in the same way as the Prog Mgr/Shell Groups command. However, it works directly with your **Windows** Program Manager group files and ignores any other shell which you may be running).

Cut and Paste Commands

You can copy program manager or other desktop shell commands directly to the **Stiletto command entry controls** used to enter commands for button configuration, launch menu, hot keys and alarms.

This is an alternative to browsing for the command when the command file name, parameters, or work directory are not known making it difficult to browse for the command.

Cut and Paste uses the P button on the command entry controls. This button will be enabled whenever there is information from a program manager or desktop shell group to be pasted: Each time you select or execute a command from the groups command, the command name and work directory are made available through the P button.

So, to paste a command, you first need to have selected the command from the program manager or desktop shell group dialog.

Follow these steps to cut and paste commands:

Start the Configuring Stiletto built-in command.

Press the Program Mgr/Shell Groups button on the Info tabbed dialog.

If needed, move the Program Manager Groups dialog box so you can access the configure dialog.

Access the command you want to configure through the button configuration, launch menu, or alarms tabbed dialogs.

Select the program manager group and command you want to paste (single click!).

Press the P button on the command entry controls to copy the information.

Save the copied command on the menu, if appropriate.

Repeat the above three steps for all commands you wish to copy.

Cancel the program manager dialog and OK close the configure dialog when done.

Resource Usage Displays

Stiletto can display free Windows resources as a button label by selecting the desired display from the Configure Button dialog.

These displays are available:

Free GDI Res	Displays the percentage of free GDI resources in the form NNg, where NN is the free percentage.
Free USER Res	Displays the percentage of free USER resources in the form NNu.
Free Memory	Displays free memory, which includes virtual memory (win16), in Kilobytes.
Min of GDI/USER	Displays the minimum of GDI and USER in the form NNf.
GDI/USER	Displays both GDI and USER free resource in the form NNg/MM, where NN is the free GDI resources (shown by the g) and MM the free USER resources.
Min and Mem (MB)	Displays the minimum of GDI and USER as well as the free virtual memory in Megabytes in the form NN/MM.M, where NN is the minimum of free resources and MM.M is the free memory in megabytes (1024K).

GDI resources include device-context handles, brushes, pens, regions, fonts, and bitmaps. USER resources include window and menu handles and related structures.

Use the Alarm Setup dialog to set the rate at which **Stiletto** refreshes the resource displays. You can also use this dialog to set a level for **Stiletto** to automatically monitor USER and GDI resources.

Changing the Order of Buttons

You can change the order of **Stiletto** buttons by clicking the New Button Order button on the Configure All Buttons dialog.

Clicking this button brings up a dialog with a row of current buttons. All 16 potential buttons are displayed. (The Configure All Buttons dialog lets you control how many buttons are actually displayed).

The top row of buttons shows the buttons in their current order. (For vertical **Stiletto** the leftmost button will be topmost). To re-order the buttons, click on them in the new order in which you want them to appear. After you click on a button, it will be moved to the bottom row showing the new button order. If you change your mind, click on any button in the new button order and it will be moved back to the old button order.

Enter command parameters here. This includes parameters for built-in commands, like the Launch Menu or the Bar Size. To prompt for command parameters, put a question mark (?) here. You can enter both parameters and a ? to prompt for parameters in the midst of a command parameter string.

Put the name of the file to be executed here, or select a built-in command using the drop down or by right-clicking on the edit box and selecting from the alphabetical list. You can type the file name directly; but normally, you use one of the Browse, Paste, or Capture buttons or drag and drop a file name from File Manager. You can use program (.exe) files or files with associated commands (eg .xls file extension for Microsoft Excel files). You can run multiple commands. If the command name starts with a semi-colon, it is ignored; this is useful for temporarily disabling commands while keeping the text for later re-use.

Press this button to get the file name for one of the programs that is currently active (executing) on your system. A menu of active windows will be displayed. Select one, and its .exe file will be copied to the edit box.

Enter the name of the working directory to use for the command here. For some Windows programs, it is necessary to set this to the directory containing configuration files, and so (starting with Version 1.9r) if you omit a working directory **Stiletto** will automatically use the same directory as the command. The work directory is also useful to set the default directory for File Opens for the command. You also use the work directory to enter the text for alarm messages or the starting directory and default extensions for Browse and Run. Finally, you can enter |filepath.wav to play the sound filepath.wav when the command is started. If you also want to use a work directory, put the |filepath.wav immediately after the work directory name. If you want **Stiletto** to use the current directory as the working directory, enter * for the directory name.

Press this button to set the working directory to the same directory as the command file is in.

Use this drop down to select the window style for the command when it is run: normal, minimized, maximized, hidden. You can also set the window to be always on top.

Press this button to browse through your directories for the file to be used.

Press this button to copy the file name corresponding to one of your program manager or other desktop shell commands (icons). To enable the button, you first have to select a command. See [cut and paste](#) commands for details.

Using Alarms to Close or Wait on Active Tasks

You can use alarms to close active tasks or to wait for active tasks to complete.

If you create an alarm with the work directory set to the string *close, then the command will be closed when the alarm rings, if it is active (nothing happens if the command is not active when the alarm rings).

If you create an alarm with the work directory set to the string *wait, then no further alarms will be processed until the command terminates (nothing happens if the command is not active when the alarm rings). You need to make sure the Ring Missed Alarms option is checked in the alarm setup dialog for *wait to work.

Close and wait alarms do **not** work with Dos programs; they only work with Windows programs.

Here is an example of the use of close and wait alarms: Suppose you have one modem which you use for two programs: a fax receiver (program /winfaxx/winfaxx.exe) and an offline mail reader with a script which automatically dials your Internet provider and downloads your mail (program /fedora/fedora.exe myscrip.t.hat). Now suppose you want the fax reader to be always running on your machine except for 2:00 AM when you want your mail reader to download your mail. Your problem is that both programs cannot be running at the same time since they both use your single modem.

The following sequence of alarms will solve your problem by closing the fax program at 1:59, starting the mail reader at 2:00, and re-starting the fax program when the mail reader finishes.

Time	Command	Work Directory	After Alarm Rings
1 59 AM	/winfaxx/winfaxx.exe	*close	Alarm again in 1 day
2:00 AM	/fedora/fedora.exe myscrip.t.hat		Alarm again in 1 day
2:01 AM	/fedora/fedora.exe	*wait	Alarm again in 1 day
2:02 AM	/winfaxx/winfaxx.exe		Alarm again in 1 day

Stiletto close and wait alarms are not intended to replace an industrial strength scheduling program. For example, they cannot handle situations where programs encounter errors or have unsaved data and so refuse to close. You are cautioned to use close and wait alarms only for non-critical data where it is easy to recover manually if exceptional circumstances arise.

Final note: if you believe that the user interface for close and wait alarms is ad hoc, ugly, and not Windows standard, you have the full agreement of the author.

Running Multiple Stiletto Button Bars

You can have many **Stiletto** button bars by running more than one instance of **Stiletto** simultaneously. For example, you could have your main button bar in the top left of the screen, and a separate button bar in the bottom middle of the screen for drag-and-drop commands like **Stiletto** rename and your own viewer, editor, or drag-and-drop file deleter. Each separate button bar uses its own configuration file.

To set up a new configuration file, start with the blank button bar file called blankini.def which is part of the **Stiletto** zip package. Copy this file to stilet2.ini (or any other file name you wish) using a Dos copy command:

```
copy blankini.def stilet2.ini
```

Now you execute **Stiletto** specifying this configuration file by creating a command for a button or on a menu which looks like this:

```
c:\stilpath\stiletto.exe stilet2.ini
```

(Use no directory name with stilet2.ini). Activate the button command or menu entry. You will get a blank, one button bar, in the middle of your screen. Click on the button and re-configure its position, number of buttons, and button contents. Don't forget to continue to have a configuration command somewhere on a button or menu. When you are satisfied with your new button bar, you may wish to assign the above command to a new icon in your program manager startup group or your **Stiletto** start-up launch menu.

If you wish, you can continue to create other button bars by creating a separate configuration file for each. All configuration files must be in the same directory as the **Stiletto** .exe file or in your **Windows** directory.

A default configuration file, stiletto.ini, is used when you run **Stiletto** without a configuration file on the command line. The name of the configuration file **Stiletto** is using is shown in the Info dialog.

Hotkeys, the desktop/title bar mouse buttons, and application sounds are determined by the first instance of **Stiletto** that requests them. Each instance will separately process chimes, alarms, wallpaper and sound randomization, and resource warnings. Hence, you should isolate your use of these features to one main instance. This will also conserve your **Windows** resources.

Stiletto will be positioned horizontally at the left of the bottom of the desktop, raised by the height of a status bar from the bottom of the screen.

Stiletto will be positioned horizontally in middle of the bottom of the desktop, raised by the height of a status bar from the bottom of the screen.

Stiletto will be positioned horizontally at the right of the bottom of the desktop, raised by the height of a status bar from the bottom of the screen

Using Icons to Label Buttons

Stiletto can display an icon as a button label; to access this feature select the desired display entry from the label drop down box in the Configure Button dialog.

These displays are available:

Icon from Left	Displays icon from left command.
Icon from Middle	Displays icon from middle command.
Icon from Right	Displays icon from right command.
Icon from File	Displays icon from file that you browse for or key in to edit box.

Controls for working with icons will appear when you select one of these entries.

If there is more than one icon in the file, **Stiletto** lets you select which icon you wish to display.

If you change the command file or icon file, you may need to press the **Show Icon** button to see the new icons from the selected file.

Stiletto normally extracts the icon directly from the file but this technique does not work for 32 bit executables if you are running in win32s (no problems with win95); if you wish to use access icons in these files, you can use an internal **Stiletto** ini option, (but due to an apparent Win3.1 bug, this will use **Dos** low memory and may lead to GPFs if you use a lot of icons).

Position Stiletto

You can position **Stiletto** manually anywhere on the desktop by using the Position **Stiletto** built-in command.

First assign this command to any **Stiletto** button and to any of the left, middle, or right mouse buttons. Then use the buttons dialog to select any standard vertical or horizontal position, depending on whether you want **Stiletto** to be oriented vertically or horizontally. Finally, click on the button with the Position command using the appropriate mouse button and drag **Stiletto** to wherever you would like it to be positioned. After you are done dragging, **Stiletto** will record its position in the configuration file and will return to this position automatically when started.

Caps Lock and Scroll Lock

The Configure All Buttons dialog contains check boxes to permit you to control the behaviour of the Scroll Lock and Caps Lock keys.

You can specify that pressing shift always clears caps lock, to avoid reversed mixed case like **STILETTO**.

You can disable the Scroll Lock key. This key is rarely used, and when activated unknowingly, causes irritating behaviour from the arrow and other keys.

You can put `DisableCapsLock=1` in the [task menu info] section of the in file to disable the Caps Lock key.

When using this features with multiple instances, make sure than only one instance uses it and hotkeys.

Use multiple menus to organize your work by collecting all your related work onto menu and attaching the menu to a button. Example: put all your Internet-access commands onto a menu. Or, if you work with a many spreadsheets, organize them into a menu/submenu hierarchy for quick access. Remember that you can use the name of the spreadsheet file as a command in the menu.

Check this box if you want **Stiletto** to simply switch-to a program if it is already running, rather than launching a new instance. Uncheck this box if you have a command line parameter that you always want processed; this will ensure that a new instance of the program is started to process the parameter.

Active Window Switching with a Button Bar

You can create a **Stiletto** button bar which automatically tracks each top-level window on your system so you can quickly switch to a new active window by left clicking a button on this button bar (the Active Bar) and close any visible window by middle-clicking (shift-left) the corresponding button. This is an alternative to the list of active windows menu item which can also be used for active window switching. The advantages of a button bar are that all active windows are immediately visible on the button bar and that switching is done with a single click. The disadvantages of the button bar are that extra resources and desktop space are used.

Note that right-clicking on a button bar can still be used to access commands, although only the leftmost button will always be visible (and so should be used for a Configuring Stiletto command or a menu containing a Configure command so that the Command is always available).

The best way to set up a button bar for active window switching is to use the `stactive.def` configuration file which is part of the **Stiletto** zip file:

Create a new copy of the `stactive.def` file called `stactive.ini`. (Create a new copy, rather than renaming, in order to keep the `stactive.def` file as a backup). You can use a **Dos** command like
`copy stactive.def stactive.ini`
to do this. Keep this file either in your **Stiletto** directory or your **Windows** directory.

Create a new icon in your Program Manager Startup or other group and set the command line to
`c:/stildir/stiletto.exe stactive`
where `c:/stildir/stiletto` should be replaced by the path to your copy of `stiletto.exe`. Call this new icon Stiletto Active Bar. (You can also assign this command to a **Stiletto** button/menu on a separate button bar from the Active Bar or put the command in your Stiletto start-up launch menu).

Double click on the new icon to start the Active Bar. By default, it will be positioned in the bottom middle of your desktop. Right click on the leftmost button and select Configure Stiletto from the menu. You can then reposition the button bar (but you cannot use the Split position). Or you can right-click on the second button and drag the Active Bar to a new position (you will need to have two windows active to do this). You can also position the Active Bar in the active window caption.

Stiletto will display a button for each top-level window of each active task, up to the maximum number of buttons set by the number of buttons displayed drop down in the buttons dialog.

Flyover help for the left button of the Active Bar displays the window title of the corresponding active window. You may want to set a small delay for flyover help for the Active Bar using the buttons dialog so this information appears as soon as you move the mouse cursor over the Active Bar.

You can use the omit list to cause any active window to be excluded from the Active Bar or to edit the name of text for the Active Bar. Or, you can use the Hide Window built-in command to hide the window.

You can cause activated task windows to be centered using a switch on the Menu Setup

dialog.

You can ask **Stiletto** to automatically minimize non-active windows

You can minimize an active bar window by this sequence of mouse keys: left down, right down, left up, right up (!).

Since the active bar will show all running tasks, you can cause minimized (iconized) non-Dos windows to be hidden using a switch on the Menu Setup dialog.

The default active bar has a menu associated with the left most button. This menu includes a n active task list submenu to allow switching to tasks which do not appear on the Active Bar because they are invisible or because the Active Bar is full.

Stiletto normally uses the middle mouse key to close active windows and the right mouse key to run commands or menus. You can switch these two with the Menu Setup dialog.

By default, the active bar will use icons to represent active windows. If you prefer text, replace the label on the leftmost button bar with any text. In general, the Active Bar will use icons if the leftmost button is assigned any of the icon labels, and text otherwise. You may also wish to using big buttons as controlled by buttons dialog. (**Dos** windows always have text labels).

The Active Bar is disabled while you are configuring it.

Normally, the active bar shows each top level window on your system. If you prefer, you can use the Menu Setup dialog to specify that only one window per active task be represented on the active bar. **Stiletto** will choose a top-level window which is not omitted through the omit list. Switching to this entry on the Active Bar will center the indicated window and bring all windows that belong to the task to the top.

It is also possible to manually create an Active Bar if you prefer to start with one of your existing configuration files. (You are creating a configuration file for a new instance). Copy and rename this file to (say) stactive.ini. Edit this ini file and make sure the line ActiveBar=1 appears under [General]. Then proceed as above. Do not forget to include a Configure **Stiletto** built-in command associated with right-clicking the first button.

Stiletto normally uses the icon in the exe file for the Active Bar. Some programs create a new icon when they run. If you wish to use this icon, there is an internal option to do this.

(Technical note: Starting with version 1.9q, by default **Stiletto** no longer uses a timer for triggering active bar refreshes. However, some user have experienced performance problems with this hook and so it is possible to return to the timer using the ActiveRefresh internal option.

Omitting Windows and Words from Active Window Lists

You can use the omit list edit box on the Menu Setup dialog to omit either words from a window name or to omit windows completely from the list of active windows or the Active Bar.

To omit a word, type the word followed by a comma. For example, you could use this technique to delete vendor names.

To omit an entire window, type the window name as it appears in the caption title of the window followed by a comma, e.g. Program Manager in the omit list will mean that no entry for Program Manager will appear.

If you include a string followed by an asterisk (*) and comma in the omit list, then any active window with caption text starting with that string will be deleted. For example, 1MBFort* will delete any program name starting with 1MBFort.

Hiding Windows

You can use the Hide Window built-in command to hide windows. You might use this if you do not want a window to appear on the list of active windows or the Active Bar.

When you execute a Hide Window command, the cursor changes to a cross. Left click on the window you wish to hide. This window, its top-level parent, and all the parents children will be hidden.

You cannot hide a **Stiletto** window or the desktop window. Some other programs will also refuse to be hidden.

If you execute Hide Window but then decide you do not want to hide a window, left click the mouse on the desktop or on a **Stiletto** window to cancel the operation.

If you want to show a hidden window, configure the **Stiletto** list of active windows to show hidden windows using the Menu Setup dialog, assign this command to a menu, then execute the menu and select the hidden window from the list.

Hints and Tips

Following is a list of **Stiletto** features you may be interested in trying. Click on the o to see help information. Also see [New for This Release](#).

Stiletto supports:

o multiple button bars.

o up to ten launch menus.

o a button bar which tracks and allows switching and closing of active windows.

o drag/drop files from File Manager to a button to start the button command using that file.

o a built-in command for single file rename which has a friendlier dialog box than File Manager rename.

o configuring a button by pressing and holding it.

o configuring a button by dropping a file on it with the ALT key held down.

o configuring a launch menu by dropping a file on a button which references the launch menu or by pressing such a key with the ALT key held down.

o you can use the menu item name to configure mnemonic keys to select a launch menu item with one key stroke. You can use these mnemonic s as hot keys for applications (ie use hot key to bring up a menu, and then mnemonic key to select item).

o positioning anywhere with the built-in Position Stiletto command.

o positioning in the caption of the active window.

o defining your own font for menus and giving menus a button look.

o quickly exiting windows with "Quick Win Exit" built-in (no confirmation unless you have unsaved info).

o built-in commands for executing files, of a type you specify from a directory you specify, either by immediately browsing for the file ("Browse and Run") or typing the file to be run ("Type and Run"); you can start the browse in any default directory.

o built-in commands for immediately starting, enabling, and disabling your screen saver.

o accessing the "middle" button commands by using shift and the left mouse button.

o putting running program windows "on top" or launching programs as "on top" (command entry controls).

o controlling whether or not and how soon flyover help and drag/drop flyover help appear.

o making label text run vertically.

o a built-in command to move the button bar quickly to a different location on the desktop.

o a built-in command which, when executed, displays a list of all built-in commands from which you can select and execute one: a way to make all of them accessible.

o a built-in Play Sound command for sound play: drop a wav file on a button with this command and **Stiletto** plays the wav file.

o a built-in Show Wallpaper command for previewing wallpaper: drop a bmp file on a button with this command and **Stiletto** shows the wallpaper.

o file associations with wav files to play them by double clicking in File Manager; file associations with bmp files to preview them as wallpaper.

o a built-in command for bring all **Stiletto** button bars to the top; assign this to launch menu 0 and you can then bring any hidden button bars to the top by accessing the menu and selecting the command. Or you can show only a specific bar by specifying that bar's configuration file name in the command parameter edit box.

o control of caps lock and scroll key behaviour.

o alarms to stop programs or wait on programs to exit.

o hotkeys to instantly access Launch Menu 0 and Program Manager groups and commands.

o resource usage displays.

o deleting phrases or entire entries from active windows lists and the Active Bar.

o prompting for command parameters by putting a ? after the command name.

o changing the order of button bar buttons.

o changing the size of buttons

o putting more than one command on a command line.

o using a command to immediately change the number of displayed buttons, add or subtract from the number of buttons displayed (eg to add a new command quickly), or to make **Stiletto** invisible.

o directly accessing and executing program manager groups and commands.

o configuring commands by drag/dropping the command file name to the edit control, by capturing running program file names, and by copying file information from program

manager.

o monitoring of **Windows** user and GDI resources.

o quick addition of reminder messages as alarms.

o use timers as button labels.

o controlling timers by the status of your modem.

o using **Stiletto** as your shell.

o running commands when **Stiletto** starts.

o playing a random sound in a random percentage of minutes (for cheap entertainment).

o If **Stiletto** is automatically becoming invisible, you can control how long **Stiletto** stays visible with the autohide delay drop down.

o you can send a sequence of keys to a running program or to a program that you start with **Stiletto**

o **Stiletto** can automatically minimize windows of inactive tasks

o commands can be assigned to hotkeys and to moving the mouse to screen corners.

o you can start, stop, and clear timers from external programs such as dialers with appropriate scripting commands

o you can create menus from program manager groups or file directories

You have successfully clicked.

Running Commands at Stiletto Startup

You can specify that **Stiletto** should run a set of commands when it starts by putting the commands you want to run on a launch menu and specifying that launch menu on the Menu Setup Dialog.

You cannot use launch menu 0 as the startup menu.

Timers as Button Labels

Using the button dialog, you can have **Stiletto** place a timer as the label on any button.

A running timer is displayed in the form **hhhh.mm** (hours, then a period, then minutes).

A stopped timer is displayed in the form **hhhhxmm**.

Timers are displayed rounded to the nearest minute, unless you specify that seconds should be shown in the Alarm Setup dialog. In this case, seconds are always shown as **.ss** at the end of the timer label.

You can select from four types of times:

Auto, Not Saved Are started automatically at 0 when **Stiletto** starts.

Auto, Saved Are started automatically at their last value when **Stiletto** starts.

Manual, Not Saved Must be started manually; not saved when **Stiletto** exits.

Manual, Saved Must be started manually; saved between **Stiletto** runs.

Use the built-in commands Start, Stop, Toggle, Clear and Reset timers to work with timers. To be effective, these commands must be associated with the button displaying the timer as a label, or a launch menu associated with that button. For example, you could place these commands on a launch menu and associate this same menu with all timer buttons on a **Stiletto** button bar.

The Reset Timer command displays a dialog box allowing you to see and optionally set the hours, minutes, and seconds associated with a timer. You also use this command to initialize newly configured timers.

Timers are updated at the same interval as the resource usage check set in the Alarm Setup dialog. But, unless you specify that seconds should be shown, the button display is only changed once per minute. You can always see the timer value to the nearest second with the Reset Timer Command.

You can have **Stiletto** play an alarm sound at regular intervals for all timers using Alarm Setup dialog. The alarm sounds two minutes before the interval to give you time to react (eg alarms every 30 minutes will sound at 28 minutes, 58 minutes, and so on).

Timers associated with non-displayed buttons are still updated, if running.

You can start and stop timers from external programs, such as dialer scripts. You can also use the external timer commands to control timers using a command launched from any button. Also see this help topic to see how to use alarms to clear saved timers daily.

You can start and stop all timers on a button bar based on the status of your modem.

You can ask **Stiletto** to log all timer events in a file.

Starting with version 1.9s, timers are normally updated by computing the difference between the current clock and the clock setting with the timer was last calculated. This method keeps timers accurate even if another application monopolizes the CPU for an extended period. However, it does mean that timers must be manually reset if you change the system time. An alternative approach which is insensitive to clock changes is to

update the timer based on assuming that timer notifications will not be lost due to other applications. You can ask for this approach with the TickTimer internaloption.

Changing Number of Buttons Displayed

You can change the number of displayed buttons on a **Stiletto** button bar by executing the built-in command Bar Size n, where n is the new number of buttons to be displayed and is placed in the command parameter edit box. For example, Bar Size with a command parameter of 12 sets the number of displayed buttons to 12. The command may be assigned to a button or a launch menu.

This command has exactly the same effect as using the buttons dialog.

If you omit the number n, **Stiletto** will prompt for it using the command parameters dialog.

You can also make **Stiletto** invisible by using Bar Size 0. In order to make **Stiletto** visible again, you need to assign the built-in command Bring Stiletto To Top to launch menu 0, attach this menu to the desktop or to the window captions using the Menu Setup Dialog, and execute this command.

You can increase (or decrease) the number of buttons by specifying +n (or -n). For example, Bar Size +1 adds one more button. You could use the feature to add a new button to which you can quickly assign a new command by (eg) dropping a file from File Manager with the Alt key held down (see button and Menu Contents dialogs).

If you have a set of different sizes you want to quickly switch among, you may want to set up a submenu of size commands on launch menu 0.

You can assign the Bar Size 0 command to the Start Up launch menu specified on the Menu Setup Dialog in order to make **Stiletto** initially invisible.

This command will be ignored while you are configuring **Stiletto** or if you put a non-numeric character after Bar Size.

Internal Configuration Options

Stiletto has a set of internal options to make technical configuration changes. To access these, you need to edit the **Stiletto** configuration (ini) file and insert one of the following options in the [General] section.

Icons from 32-bit exe files: normally, **Stiletto** reads icons directly from exe files rather than using the standard **Windows** routine ExtractIcon, as for Win3.1x this routine apparently has a bug which causes excess Dos (low) memory to be used which can lead to not enough memory to run program errors. But the **Stiletto** internal routine does not work with 32-bit exe files accessed from win32s. If you want to access icons in these files, you can try putting UseExtractIcon=1 in the **Stiletto** ini file [General] section. However, you may get GPF or other Windows errors. Note: this bug has been fixed in Win95 and so **Stiletto** always uses ExtractIcon for Win95.

Suspended flyover help: After you launch a command, flyover help is normally suspended until you move the mouse off the **Stiletto** button bar. This is to prevent problems with full screen **Dos** commands: if flyover help is not suspended, flyover help will appear unnecessarily after you activate a full screen **Dos** command and minimize the **Dos** window. But, if you never use full screen **Dos** and want flyover help not to be suspended, you can put FlyoverSuspend=0 in the **Stiletto** ini file [General] section.

Class Icons for the Active Bar: **Stiletto** normally uses the icons from the exe file for the active bar; some programs create new icons for their class windows when running. If you prefer to use these icons, put UseClassIcon=1 in the **Stiletto** ini file [General] section.

3D Dialogs: If you do not want them, put 3D=0 in the **Stiletto** ini file [General] section.

Fast Menu Appearance on Click: When you click on a button to display a menu, **Stiletto** will display the menu on the click: you can then roll to the selected item and release to select an item or release and click again on the desired menu item. If you prefer the second method, you have to be careful not to move the mouse before releasing (as this is taken as a no selection under the first method). If you always use the second method of accessing menus, you can use FastMenu=0 in the [General] section of the ini file to turn off the first method so you don't have to be careful about moving the mouse.

Time Delay for Hiding Program Manager: If you check Hide Program Manager on the Menu Setup dialog, **Stiletto** will wait 5 seconds after starting by default to hide program manager. You can change this by specifying HideProgManDelay=n in the **Stiletto** ini file [General] section, where n is the number of milliseconds to delay (eg HideProgManDelay=1000 for one second).

Time Delay for Auto-Repeat Buttons: You can change the interval at which the autorepeat buttons (eg in the alarm dialog) repeat by specifying StrobelInterval=n in the **Stiletto** ini file [General] section, where n is the number of milliseconds between repeats while the button is held down.

Separator character for multiple commands: You can change or eliminate the character used to separate multiple commands to the character "c" by specifying CommandSepChar=c in the **Stiletto** ini file [General] section, where c is the new separator character, which cannot be an alphanumeric, comma, or space. To eliminate command separation, specify CommandSepChar=0.

Owned windows in active task lists: Starting with version 1.9q, **Stiletto** no longer displays "owned" top-level windows in active task lists. This cuts down the size of such lists. But if

you prefer to see these windows, specify ShowOwned=1 in the **Stiletto** ini file [General] section.

Transparent button bar: You can ask for a transparent buttons bar by putting transparent=1 in the **Stiletto** ini file [General] section and making sure that Use Windows Button Colours in the Configure All Buttons dialog is unchecked. This may cause some strange (visual only) side effects, especially with multiple **Stiletto** instances, and may lead to a **Stiletto** GPF when used with some full screen Dos programs, so it is being offered only as a "use at your own risk" internal option. An active bar cannot be made transparent.

Rotated text: Starting with version 1.9t, if you want rotated text on button labels, you must use the internal option rotatetext=1 in the **Stiletto** ini file [General] section.

Active Refresh: Starting with version 1.9q, **Stiletto** uses a hook rather than a timer to check for the current active window for the active bar and for active caption placement. This may cause delays when moving or resizing windows, so you can return to the timer by specifying ActiveRefresh=n in the **Stiletto** ini file [General] section, where n is the number of milliseconds between refreshes (eg ActiveRefresh=1000 for refreshes once per second). You must set n larger than 100.

CheckStretch: **Stiletto** normally checks to ensure that your video device driver can perform the stretch operation needed to display icons. However, some drivers, although able to perform the stretching, report that they cannot. To ignore this false report, specify CheckStretch=0 in the **Stiletto** ini file [General] section.

TickTimer: Starting with version 1.9s, timers are normally updated by computing the difference between the current clock and the clock setting with the timer was last calculated. This method keeps timers accurate even if another application monopolizes the CPU for an extended period. However, it does mean that timers must be manually reset if you change the system time. An alternative approach which is insensitive to clock changes is to update the timer based on assuming that timer notifications will not be lost due to other applications. You can ask for this approach by putting TickTimer=1 in the ini file [Alarm Info] section.

LongFileNames: Put LongFileNames=0 in the [General] section of the **Stiletto** ini file to disable long file names support for Windows 95.

WideBrowse: Put WideBrowse=0 in the [General] section of the **Stiletto** ini file to disable the wide browse file dialog used for long file names with Windows 95.

Win95: Put Win95=0 in the [General] section of the **Stiletto** ini file to disable Windows 95 support.

NoSoundSave: Put NoSoundSave=1 in the [General] section of the **Stiletto** ini file to disable changes to the sound settings (for Windows 95).

AutoDisplayChange: For Win 95, if dynamic autosizing conflicts with the operation of your screen driver (eg for virtual desktops larger than the screen size), you can disable it by putting AutoDisplayChange=0 in the [general] section of you Stiletto.ini file.

Controlling Timers Externally

It is possible to start, stop, and clear timers from external programs, such as dialers, if these programs can execute windows commands. For example, the Trumpet win sockets program script language can do this with the exec command, which looks like this: `exec "c:\exepath\exename.exe parameters"`.

If your dialer does not support external commands, you can run both the **Stiletto** timer command and your dialer as multiple commands to start the program.

The instance of **Stiletto** with the timers to be accessed must already be running. Also, since more than one instance of **Stiletto** can be running, you need to specify the configuration file for the instance with the timer label that you want to reference.

To control a timer externally, you execute a `stiletto.exe` command as if you wanted to start a new instance of **Stiletto** with a command line that has three parameters:

1. The name of the ini (configuration) file; use `stiletto.ini` if you are running only one instance.
2. One of clear, start, or stop.
3. The number or timer id of the button with the timer. You can assign a single, non-digit character (using the button dialog) as the id of a timer so you do not need to change external command references if you re-order buttons. Or, you can use number 0 to refer to the last timer button from which a command was executed.

For example:

```
c:\path\stiletto.exe stiletto.ini stop 1
```

stops the timer on button 1 for the **Stiletto** instance with configuration file `stiletto.ini` (the default if you do not specify a configuration file).

```
c:\path\stiletto.exe stilet2.ini start 0
```

starts the timer on the last button with a timer label from which a command was executed for the **Stiletto** instance with configuration file `stilet2.ini`. The idea is that the dialer (or other) program would have been launched from this button or a menu attached to this button. Finally, the command:

```
c:\path\stiletto.exe stiletto.ini clear x
```

clears the timer associated with the button with the timer id "x".

You can also use this technique from within **Stiletto** to clear, stop, or start a timer on a button which is different from the button from which the timer command was launched: To do so, assign a command of the form given above to a button or menu within **Stiletto**.

Remember that **Stiletto** will still start and stop timers on undisplayed buttons. For example, you can set up a **Stiletto** button bar with a visible button manual-saved timer to show daily totals and a non-displayed button (say 16) to show monthly totals. Attach a launch menu of network related commands to the displayed button and add timer control commands (for back-up to dialer script) to this menu. Put two timer starts in your dialers

login script, and the corresponding two timer stops in your dialers by script. Put dynamic sizing commands on the menu to show and hide the monthly total button on timer 16.

To automatically clear a saved timers once per day, set up an alarm with these characteristics (using button 1 for example):

Time: 12:01 AM

After alarm rings: Alarm again in 1 day

Command: c:\path\stiletto.exe stiletto.ini clear 1

The "Ring Missed Alarms" checkbox on the Alarm Setup dialog must also be checked for this to work (unless you start **Stiletto** each day at 12:01!). You can use a similar technique to clear timers once per month (ring on first of month at 12:01)

You can play a sound whenever you execute one of these commands by placing | filepath.wav at the end of the command line, for example:

c:\path\stiletto.exe stiletto2.ini start 0 |c:\win\tada.wav

starts the timer on button 0 and plays the sound file c:\win\tada.wav. You need to provide the entire file path, including the .wav, and you **cannot** place any blanks after the |.

You can also control timers via the status of your modem.

You can also start and close an instance of **Stiletto** from your dialer if it supports command execution from scripts.

Adding Reminder Message Alarms

You can quickly set up a reminder message to appear at a specified time with the built-in command Add Reminder. This command displays a dialog box which provides a short cut way to add an alarm to display a message box.

To enter a message date, enter the year, month, and day as one or two digits separated by a space. When you enter a valid date, the day of week and date will be displayed beside the date you enter. You can use the "=" button to reset to the current date, and the "+" and "-" buttons to change the year, month, or day. (These buttons will repeat if held down.)

To enter a message time, enter the hour and minute as one or two digits separated by a space. Use a 24 hour clock or put a (for AM) or p (for PM) after the time. When you have entered a valid time, it will be displayed in AM/PM format beside the time you enter. You can use the "=" button to reset to the current time, and the "+" and "-" buttons to change the hour or minute. (These buttons will repeat if held down.)

After you have entered a valid time, date, and at least one character in the message edit box, the "Add" button will be enabled and you will be able to save the reminder message.

You can use the After Alarm Rings drop down list to have the message automatically re-displayed at regular intervals.

You can specify that the message should be displayed using the Add Reminder dialog (rather than a plain Message Box) which will give you more flexibility in choosing the next alarm date/time if you decide to recycle the alarm.

Positioning Stiletto in the Active Window

You can position **Stiletto** at the left or right of the active window caption by selecting "L Cap" or "R Cap" as the position in the Buttons dialog.

You can have up to two instances of **Stiletto** positioned in an active caption: one at the left and one at the right.

You can position an active bar in the caption.

When there is no active window to use for the position, or if the active window caption is too small, **Stiletto** will move to a default position on your desktop. To set the default, position **Stiletto** manually using the position command. **Stiletto** will use the last position set by this command as its default.

Stiletto normally uses a system hook to change position as the active windows changes, but if this causes performance problems for you then you can try a timer-based check using an internal option for active refresh.

Changing the Size of Stiletto Buttons

You can change the size of **Stiletto** buttons using the width and height magnify percentages on the Buttons dialog and the Configure Button dialog. **Stiletto** sets the base button size using the Buttons dialog and then modifies this size for each particular button with the percentages from the Configure Button dialog.

These numbers are percentages: setting them to 100 yields the default button size. You can experiment with any other value between 10 and 999.

If you choose too small or too large a value, you may find that you can no longer access the button with your Configure Stiletto command. As long as you can still access any button, press and hold it and then assign the configuration command to that button so you can reset the percentages. Or, another way to solve this problem is to edit the stiletto.ini configuration file; look in the [General] section and set Ymagnify=100 and Xmagnify=100.

Using Stiletto as Your Shell

It is possible to use **Stiletto** as a partial replacement for your shell. **Stiletto** will handle your day-to-day shell needs and so save you the resources of starting program manager most of the time.

To make **Stiletto** your shell, use the change shell button, accessible from the Info dialog, or edit your system.ini manually. (If you manually edit your system.ini, make sure you comment out your old shell= with a semi-colon so that it is easy to restore your old shell).

Note: for **Stiletto** to function as the shell, the stilethk.dll file must be in your Windows directory or on your file path. The newest version of the ssetup program does this automatically, but if you have not used this program to install your latest upgrade to **Stiletto**, you will have to manually copy stilethk.dll to your Windows directory, or let the change shell dialog do it for you.

Stiletto has two constraints when used as a shell:

First, it cannot handle requests from other programs (usually setup or installation programs) to view or add program manager icons and groups. Its best to remember to start program manager before running a setup or installation program (just run progman.exe from your Windows directory; it need **not** be your shell). But if you forget, and such a request is made, **Stiletto** will display an error dialog and give you the option of closing the offending program and also starting program manager. Often simply starting program manager will allow the program to continue. But some setup programs will lock your system if they get no response to their requests for program manager information. After program manager is started, you can retry the operation that caused the problem, if the operation does not complete automatically. You can shut down program manager when it completes (this will not shut down Windows since **Stiletto** is the shell).

Second, **Stiletto** cannot supply unique icons for Dos programs. Any minimized Dos box will have the default blue MS-DOS icon.

When **Stiletto** is used as your shell, it will properly start any programs on the run= and load= lines in your win.ini file (separate commands by blanks, not commas or semi-colons). However, you must use the **Stiletto start-up** launch menu, not the program manager StartUp group, to start other programs.

When Stiletto is used as the shell, the built-in Prog Mgr Files command is automatically used to display program manager icons and groups (even when the program manager /Shell Grps command is referenced, since this latter command depends on Program Manager running).

If your try to exit the instance of **Stiletto** which is acting as your shell, **Stiletto** will first confirm that you want to exit Windows.

Change Your Shell

To change the shell, use the Change Shell button accessible from the Info dialog.

The dialog shows you the file name of your current shell and provides an edit box to enter the file name for a new shell. You must enter the complete file path, including all directories and the file extension (.exe). Or you can press the button to set the edit box to the file path for **Stiletto** or the file path for your preceding shell (if you used this dialog to set **Stiletto** as your shell previously).

Pressing the Change Shell button saves the new shell name. It will take effect the next time you start Windows.

Pressing the Change Shell and Restart button saves the shell and restarts Windows. Any configuration changes will not be saved, so make sure you have used the Apply button first.

For **Stiletto** to be your shell, stilethk.dll must be in your Windows directory; if it is not, **Stiletto** will offer to put it there for you before changing the shell.

See change the shell for further information.

Timer Logs

You can ask **Stiletto** to log timer events by using the Keep Timer Log check box on the Alarm Setup dialog.

The log file will have the same name as the configuration file used in the instance of **Stiletto**, except that the file extension will be .tlo. For example, the log file for the default configuration is stiletto.tlo. The log is always placed in the same directory as the **Stiletto** ini file.

A log file entry will be written whenever a timer starts, stops, or is re-set. As well, when **Stiletto** shuts down, a stop timer entry will be written for any running timers. When **Stiletto** starts up, a start timer entry will be written for any automatic start timers.

The logs have fixed-format records structured as follows

Column	Contents
1	Always blank.
2- 3	Button of last timer command.
4	Always blank.
5	Timer id (single character).
6	Always blank.
7	Action: "+" if timer started, "-" if timer stopped, "0" if reset
8	Always blank.
9-12	Year when event recorded.
13	Always blank.
14-15	Month.
16	Always blank.
17-18	Day.
19	Always blank.
20-21	Hour (military clock, ie 24 hour time)
22	Always blank.
23-24	Minute
25	Always blank.
26-27	Second
28	Always blank.
29-36	Total timer value in seconds.
37	Always blank.
38-42	Whole hours in the timer.
43	Always blank.
44-45	Whole minutes in the timer.
46	Always blank.
47-48	Seconds in the timer.

To be clear: the timer value is shown in two different formats: columns 29-36 show the timer value in seconds. Columns 38-48 show the timer value as hours, minutes, seconds.

I tried to find more information so as to extend these fixed-length records to 80 characters (to avoid wasting punched card space), but could not do so.

Running Multiple Commands

If you need to associate more than one command with a button, the cleanest way is to use a Windows batch command program and run the batch file from **Stiletto**. One you should consider is **batsh**; it is freeware available from:

Thomas Nyffenegger, FMI, PBX 2543, 4002 Basel, CH
Email: nyffenegger@fmi.ch
Web: <http://www.fmi.ch/groups/ThomasNyffenegger/Group.html>

It is also available from SimTel ([ftp.coast.net](ftp://ftp.coast.net)).

If you only need a quick (and dirty) way to run more than one command, you can do so by separating multiple commands with a less than sign (<) in the command entry box and parameters edit box of the command entry controls.

Put the first command name in the command entry box (do not use a long file name), and put its parameters and the remaining commands and parameters in the parameters edit box.

Alternatively, you can put all the commands in the command edit box, separated of course by the < character..

You need to manually type in the command file names. It is also possible to use built-in commands, but you have to type the internal name. (It starts with a *: to find one for an external command, create a dummy entry for the desired name and look in the ini file. Two important ones are *wait for delays and *send to send keys).

For example:

```
c:\path\dial\dialer.exe<c:\stiletto\stiletto.exe stiletto.ini start t  
starts program dialer.exe and starts timer t.
```

You can insert pauses between commands with *wait n, where n is a single digit delay in seconds (*wait 0 waits for 1/3 of a second). This is mainly useful with Send Keys built in command, eg

```
c:\prog\splash.exe<*wait 4<*send ** "h e l l o"  
starts the program splash.exe, waits 4 seconds, and then sends the key sequence "hello".
```

There are situations where you only want to execute a sequence of commands when a preceding command is launched initially, not when it is switched to if active (example: sending a series of keys at start up). The built-in command *abifa (abort if active) will stop the processing of a series of multiple commands if a preceding command was switched to, instead of being launched.

If the default separator character causes you problems, or if you want to disable multiple commands, you can do so with an internal configuration option CommandSepChar.

You cannot use long file names in multiple commands.

Closing, Minimizing, or Maximizing an Active Window

You can close, maximize, or minimize a running task with a command of the form
commandName command windowId

where

commandName is either c:\yourpath\stiletto.exe (useful for access from windows batch files) or the builtin command Stiletto Exec (*sexec in multiple commands)

command is **close** or **min** or **max**

windowId is either the path name to the executable with the window to be closed, or an asterisk followed by the caption text of the window. For captions you can optionally replace some ending characters by using an asterisk: eg *PIF* finds the first window with a caption starting with PIF. Or you can replace beginning characters: eg **NotePad to find a caption ending in NotePad. If the caption contains blanks, enclose the windowsID in quotations marks, eg "*PIF Editor*".

If you use the builtin command Stiletto Exec, you put the command and windowID parameters in the parameter edit box.

To put this command in a command string, use *sexec:

```
<*wait 2<*sexec max "*My Caption"
```

waits 2 seconds then maximizes window with caption "My Caption".

This command is intended for use in batch or script files although it can also be assigned to a **Stiletto** button or menu. See also close an active window and close or wait for running tasks.

For example:

```
c:\stilpath\stiletto.exe close c:\work\mysheet.xls
```

closing any application associated with the document mysheet.xls (usually Microsoft Excel).

Stiletto uses a safe close: if there is unsaved information the application will prompt you before closing.

You can close an instance of **Stiletto** with a command of the form

```
c:\stilpath\stiletto.exe stilett2.ini close
```

where "stilett2.ini" is the file name of the configuration file used to start the instance of **Stiletto** you wish to close. Note the the word "close" is placed at the end. This command could be useful in dialer scripts to start and a stop a specific button bar with the network access.

Browse and Run Command

The Browse and Run built-in command displays a modified version of the file open common dialog box. You can search for a file or document and select it (eg by double clicking) to have that file executed. As discussed in more detail below, you can use this command to show all files of a certain type in a directory for you to select one to execute. For example, you could have a menu entry called "My Spreadsheets" which executes a Browse and Run command showing all .xls files in a default directory that you specify.

Stiletto will follow any file associations; for example, double clicking on a .txt file will start your default text editor (often NotePad). You set file associations with the File Manager File|Associate.

You can enter command parameters in the appropriate edit box in the dialog.

Stiletto remembers the last 25 commands you execute; you can select one of them from the drop down box.

You can set both the default starting directory for the search and the default file types displayed in the file list box using the Work Directory of the Browse and Run command entry controls. Using this feature, you can tailor Browse and Run commands to, for example, go immediately to the directory where you keep your Excel spreadsheets and list all the spreadsheets so you can execute one.

To set the starting directory, type it in the work directory box.

Set the default file extensions as follows: After the default start directory (if used), type a string that consists of a <, followed by the name of the file types, followed by another <, followed by the default extension in the form *.ext. For example:

```
c:\mysheets<Excel Files<*.xls
```

sets the starting directory to c:\mysheets and fills the file list box with all files ending in .xls. You can specify more than one file extension by separating them with semi-colons; for example:

```
c:\<Editable Files<*.txt;*.ini
```

sets the default directory to c:\ and lists all ini and txt files. You can also work with more than one type of file:

```
c:\<Word Files<*.doc<All Files<*.*
```

sets the starting directory to c:\ and fills the file list box with .doc files. You can select All Files when working with the dialog.

New Features

Following is a list of new **Stiletto** features for each version. Click on the o to see help information.

V1.9u1 (Mar 96)

- o create menus from files directory trees or Program Manager groups.
- o added on top and not on top entries to commands which apply to the window beneath the mouse cursor.
- o You can ask Stiletto to use a larger font for flyover help.
- o right-click on command entry controls to get menu of built-in commands, sorted alphabetically

bug fixes

- timer logs now work properly for timers with no ids
- Win95 screen display resolution changes now handled automatically
- message text now shown for alarms for "Add Reminder Msg" command
- maximized windows activated properly
- handle no wallpaper properly
- 1.9u1 fixes resource leak related to flyover help

V1.9t (Jan 96)

- o **Stiletto** now supports Windows 95 long file names in browse dialogs
- o you can send a sequence of keys to a running program or to a program that you start with **Stiletto**
- o **Stiletto** can automatically minimize windows of inactive tasks
- o **Stiletto** can automatically hide windows of inactive tasks
- o you can insert pauses in multiple command sequences with *wait.
- o mouse click for launch menu 0 expanded, including support for Win95
- o built-in commands are prefixed with an asterisk (*) to avoid conflicts with long file names in Windows 95.
- o if **Stiletto** is automatically becoming invisible, you can control how long **Stiletto** stays visible with the autohide delay drop down
- o You can have **Stiletto** automatically set the number of displayed buttons to the number of used buttons.
- o you can now put any command on start up menus
- o active bars positioned in centre of screen move to stay centered as number of active tasks changes
- o added builtin command to minimize window from list of active windows
- o added Restart Stiletto built-in command to restart **Stiletto**, possibly with a new ini file
- o added Close/Min/Max/Hide window under cursor commands; useful eg for launch menu 0
- o put "sort" as a command parameter to sort built-in commands list when the dialog opens
- o added ability to Restart an instance of **Stiletto** with an new ini file using an external command (eg from a batch file)
- o add ability to close/minimize running programs from a **Stiletto** command or by using a builtin Stiletto Exec command.
- o add ability to copy or delete files from a **Stiletto** command or by using a builtin Stiletto

Exec command.

V1.9s (Dec 95)

- o you can specify a delay for screen corner commands
- o added Clear Timer and Toggle Timer built-in commands
- o added Execute Dos Command to execute Dos command and then restart Windows
- o you can show seconds on timer labels; as well, the method of updating timers has changed

Bug Fixes:

- fixed problems with dd-mm time/date
- fixed problems with icons for Win95 non-exe files
- if alarm rings while typing, the alarm message is no longer over-keyed
- added digit '3' to hot keys
- fixed wallpaper dialog for case when wallpaper is "(none)"

V1.9r2

- o commands can be assigned to hotkeys and to moving the mouse to screen corners. (You must manually re-assign hot keys assigned to the program manager shell or launch menu 0 in previous versions)
- o you can automatically hide **Stiletto** after each command is executed
- o you can specify that **Stiletto** be unhidden when the mouse cursor bumps the screen edge nearest the bar position
- o **Stiletto** has been made Windows 95 aware and compatible
- o you can specify that right-clicking on a specified part of the title bar closes the associated window (win3.1x)
- o you can start the configuration dialog by left-clicking on the bar with the Ctrl key pressed.
- o you can minimize all windows with the Minimize All command
- o you can restart all **Stiletto** buttons with the Reset All **Stiletto** command (useful eg for Win95 after you dynamically change the display configuration)
- o you can insert a list of the 10 most recently run commands/documents into a menu with a built-in command (these commands need not have been started by **Stiletto**)
- o the Windows Shutdown built-in command activates the standard windows/win 95 exit dialog
- o you can hide all or a specific instance of **Stiletto** with the Hide **Stiletto** command
- o you can combine time and date on one button face
- o there can now be up to 50 menus and they can be given meaningful names
- o the command parameters are now entered in a separate edit box, rather than after the command name
- o if the working directory is left blank, it now defaults to the same directory as the command
- o commands starting with a semi-colon (";") are ignored: useful to temporarily disable alarms or other commands.
- o you can specify that the computer beep be used for the alarm sound by using an alarm sound filename of *
- o **Stiletto** alarms are no longer wrongly affected by Daylight Savings Time (you may have to manually adjust your current alarms)
- o you can keep a file log of rung alarms

bug fixes: 1.9r3 fixes a bug which prevented hot keys involving Alt from properly launching menus

1.9r4 fixes a bug involving windows shutdown and certain TSRs

1.9r4 fixes a bug involving size 0 and Bring To Top

For 1.9qa:

o you can set individual button face and text colours (1.9q2)

o you can ask for a Win95 or OS/2 look for menus (1.9q2)

o you can specify window captions of icons never to be hidden (1.9q6)

o you can use an internal option to specify no sound output for win95

o you can use an internal option to turn off checking for stretch icon capabilities

o you can close an instance of **Stiletto** from a command line

o **Stiletto** can now be positioned in the caption of the active window

o the **Stiletto** button size can be changed; also see button dialog

o you can specify vertical or rotated text for button labels

o you can specify the font for menus and you can ask for a button look for menus

o you can start new columns in menus with the built-in command New Menu Column

o you can configure a menu by dropping a file on a button which launches the menu, or by pressing a button which launches the menu with the Alt key held down

o added built-in Configure commands to go immediately to any tab

o added built-in screen saver start, enable, disable commands

o you can now use Stiletto as a (partial) shell replacement

o you can now start and stop timers based on whether your modem is connected

o active bar now uses hook, not timer, for faster response (but you can revert to using a timer though internal option if you experience delays with moving/sizing windows)

o when using an active bar, you can automatically hide minimized (iconized) windows

o using a built-in command, you can stop windows being always on top.

o added built-in Type and Run command so you can type and run commands

o improved Browse and Run format and made it possible to specify default file extensions as well as starting directory

o changed maximum label length on buttons to 31 characters

o new date formats allow day of week to be included

o alarm and reminder dialogs now have buttons to scroll time and date

o alarm reminder messages have improved format and can be changed when rescheduling an alarm after it rings

o you can now play a sound when executing external timer commands

o you can ask **Stiletto** to log timer events

o if you drop a file on a command, and there is no default work directory, **Stiletto** uses the

directory of the dropped file for the work directory

o if you drop a file, Stiletto always attempts to start a new instance of the command with the dropped file

o you can increase or decrease the relative size of the button bar dynamically with the Size +n and Size -n commands.

o you can now run several commands at once

o you can now close running tasks from the Stiletto command line

o you can now put Launch Menu commands in menus

o you can have a transparent button bar

o the stilethk.dll is now stored in the appropriate windows directory

For 1.9m:

o new built-in Add Reminder command allows reminder message alarms to be quickly added

o you can specify that alarms should be recycled after ringing for the next week day

o you can force menus to appear only after click AND release by putting FastMenu=0 in .ini file in [General]

For 1.9j:

o you can start, stop, and clear timers from external programs such as dialers with appropriate scripting commands

o you can drop files into the middle of a command line by putting a | at the position where the dropped files are to appear

For 1.9i:

o added Size command to quickly change number of buttons displayed:

o putting UseClassIcon=1 in .ini file in [General] section causes icons established by program window class to be used rather than icon in exe file

o putting FlyoverSuspend=0 in .ini file in [General] section causes Flyover help to appear after you launch command if you leave mouse on button bar (will cause problems with fullscreen Dos)

For 1.9h:

o added optional 1 window only per task on active bar

o putting UseExtractIcon=1 in .ini file in [General] section causes ExtractIcon to be used for icons: works for 32 bit executables but uses more Dos memory and may cause GPFs.

o added auto-backup option for saved timers (see alarm info tab)

o added built-in Bring Stiletto to Top command

o added file association support for wav and bmp; associate **Stiletto** with file type in File Manager

o added ability to access launch menu 0 by shift-clicking anywhere on a window

o added stopwatch timers as labels

o added control of text colour for button bar and flyover help

For 1.9:

o can now use a button bar to track and switch active tasks

o added ability to specify that if a command is already running when **Stiletto** is about to launch, then the active command is switched-to

o added ability to center window of switched-to active tasks

o added ability to specify help text for flyover help

o added ability to manually specify swapping of mouse labels

o added ability to set delay for flyover help

o flyover help window now moves with cursor over Stiletto bar

o increased maximum number of entries per menu to 200

o added ability to specify a sound file to play when a new instance of a command is launched

o added built-in Hide Window command to select and hide windows or icons

o added drag/drop Show Wallpaper built-in command

o added "hints and tips" section to help

o added ability to run commands at Stiletto startup

o if Program Manager is hidden, it is shown when the last instance of Stiletto exits

o **Stiletto** now checks in the windows directory for the ini file if the file is not found in the **Stiletto** directory

o you can have Stiletto prompt for command parameters by following the command name with text that includes a ?

bug fixes:

stopped using Dos (low) memory for Icons

stopped lower-casing command parameters

compiled using 286 instructions only

improved omit list description in help

rename, browse/run, Prog Mgr/Shell dialogs now appear near launching button bar

fixed GPF when adding to empty launch menu 0

for 1.9a, Icon from Left/Right now works correctly for swapped mouse buttons

for 1.9b, click for launch menu 0 now works for any combination of title and window

settings

for 1.9g, fix display of timers

for 1.9h, switch to active task activates last popup

for 1.9k, alarms now checked for during first minute

for 1.9m, fixed bug which sometimes trashed menus

for 1.9m, accessing launch menu from built-ins list caused GPF

for 1.9m, fix menu positioning from button bars at bottom right

for 1.9q, keys work on hot-keyed menu 0 even if Stiletto not on top

for 1.9q, fixed problems with cancel in configure

for 1.9q, fixed problem with [in menu names (but you cannot use ´ in menu names

now)

for 1.9q, fixed problems with key navigation of combo boxes

for 1.9q, improved tab sequence on some dialogs

for 1.9q, fixed mistaken active switch to invisible window

for 1.9q, you can now use "," in command line and "[" in menu item name
for 1.9q8, fixed positioning bug for menus launched from other menus

V1.8

o added optional icons on button labels: see help for buttons

o added multiple launch menus: see launch menu in help

o you can now position Stiletto manually (ie floating button)

o added capability to control caps lock and scroll lock key nasties

o added drag/drop button configuration: see button config help

o added drag/drop play sound command: see Play Sound in built-in help

o you can now specify button colour: see button tab in configure

o added hot key to bring up program manager command access:

o made narrow vertical Stiletto OK if no time/data/resource

o added built-in commands and key/menu options to show/hide Progman:

o added history of commands run and command parameters to Browse/Run

o added Apply button to configuration tabbed dialog (saves changes made so far even if Cancel subsequently used)

o made flyover help menu topmost so it is always visible-

o added config file name to task name for clarity

bug fixes:

button labels on all buttons dialog now updated after change

cleaned-up up some duplicate data in separate instances

dropped "flyover timeout with no button" debugging message

mouse hooks caused problems if 1 of many instances ended

wallpaper/windows now re-painted if Taskman ends Stiletto

V1.7

o added multiple button bars

o added split button bar: see Buttons Configure

o added 3 new bottom of screen positions: see Buttons configure

o added hot key to bring up launch menu

o added upper case option for flyover help: see Buttons configure

o increased maximum number of buttons to 16

o new name for ini file in installation zip to avoid over-writing existing ini during installation

V1.6

o added alarms that close tasks and wait for tasks to complete

o added left/right label swap when mouse buttons swapped

V1.5

o added Capture button to get file name from active tasks
o added capability to re-order buttons
o added common Windows applets to built-in commands list

Bug Fixes:

flyover help caused full screen Dos box to minimize
clarified help on command entry controls and added
hot spot graphic with explanations

V1.4

o added Prog Mgr/Shell built-in command to access command groups from other shells in addition to ordinary Program Manager groups
o added optional capability use Windows button colours for button bar
o added ability to drag/drop files for configuration of buttons, menu, sound, paper

Bug Fixes:

error message dialog box wasn't always modal
made it possible to drop more than one file to the button bar
got rid of bolding on Flyover help text
changed name "task menu" to "launch menu" in help
improved "randomness" by using better starting seed

V1.3

o added resource usage display
o added ability to set level at which low resource warns
o added ability to control interval between checks for low resources

Bug Fixes:

changed setup.exe to ssetup.exe to remove conflict with Windows setup

V1.2

o added Program Manager groups (via Files)
o added flyover help
o added drag/drop and drag/drop rename built-in- Bug Fixes:

Controlling Timers by the Status of the Modem

It is possible to start, stop, and clear timers based on the status of your modem. Currently all timers on a button bar must be controlled simultaneously.

To start timers when your modem is connected, assign the following command to a button or menu and execute it:

```
comstart n
```

where n is the COM port number, eg 3 if your modem is on COM3. To stop timers when your modem disconnects, specify

```
comstop n
```

After this command is executed, **Stiletto** will wait for the modem to be connected and then stop all running timers when the modem disconnects.

If you want to both start and stop timers, specify

```
comboth n
```

By combining this with the execution of multiple commands, you can put your timers under modem control and start your modem communication program:

```
comboth 2 <c:\comm\commprog.exe
```

Assuming < (the default) is the command separation character, this command puts timers under control of the modem on COM2 and then starts the program commprog.

You can also execute these commands from external programs such as dialer scripts.

The instance of **Stiletto** with the timers to be accessed must already be running. Also, since more than one instance of **Stiletto** can be running, you need to specify the configuration file for the instance with the timer label that you want to reference.

To control a timer externally, you execute a stiletto.exe command as if you wanted to start a new instance of **Stiletto** with a command line that has three parameters:

1. The name of the ini (configuration) file; use stiletto.ini if you are running only one instance.
2. One of comstart, comstop, comboth (no *).
3. The number of the COM port of the modem..

For example:

```
c:\path\stiletto.exe stiletto.ini comstart 1
```

puts the timers for the **Stiletto** instance with configuration file stiletto.ini (the default if you do not specify a configuration file) under modem control. If your dialer supports script files, you can combine external timer commands with modem control commands, like this:

```
c:\path\stiletto.exe stiletto.ini comstop 1
```

```
c:\path\stiletto.exe stiletto.ini start x
```

With these commands in the login script file for your dialer, timer x will be started when the script file is executed and all timers will be stopped when the modem on COM1 disconnects.

Hot Keys and Screen Corner Commands

The Keys/Mouse dialog is displayed when you click on the Keys/Mouse tab of the Configuring Stiletto dialog.

Use this dialog to set commands associated with Hot Keys or with moving the mouse to one of the screen corners.

The left side of the dialog shows a list box containing all hot keys/screen corner commands.

The right side of the dialog contains a set of controls for entering or changing a single hotkey including modifier keys (shift, ctrl, alt: not used for mouse corners), a drop down box to select the hot key or screen corner, and a set of command entry controls to enter the command to be executed.

In the middle of the dialog are a set of buttons which are only enabled when the corresponding action can be performed. For example, the Replace button is only enabled when you have entered a valid hot key, command, and selected an existing key from the list to be deleted.

You must select at least two of the modifier keys (shift, ctrl, alt) for hot keys to be valid and the Add/Replace buttons to be enabled.

You need to specify Shift explicitly to access shifted characters; eg on a North American keyboard you access Ctrl-Alt-! via Ctrl-Alt-Shift-1.

You can assign more than one command to a hot key by repeating the hot key in the list.

You can specify a delay for the screen corner commands; the command will only be executed if you leave the mouse cursor in the corner for at least the specified delay.

To add a new hot key:

- Check at least two modifier keys (not needed or used for screen corners).
- Select the key or screen corner.
- Enter a valid command.
- Press the add button.

To replace or change an existing hot key:

- Select the hot key in the list box.
- Press copy or double click.
- Check at least two modifier keys (not needed or used for screen corners).
- Select the key or screen corner.
- Enter a valid command
- Select the hot key to be changed from the list box.
- Press the Replace button.

To delete a hot key:

- Select the hot key from the list.
- Press the delete button.

Windows 95 Compatibility

Stiletto works well under Windows 95; following are more details on **Stiletto** running under Windows 95:

Long file names are supported and "widened" browse file dialogs are used (as a Win16 program, it is possible but difficult for **Stiletto** to access the new Win95 dialogs). You can disable widened dialogs with the WideBrowse internal option; you can disable long file name support entirely with the LongFileNames internal option.

(Long file names are incompatible with multiple commands. Also, although **Stiletto** will browse for long file names for sounds and wallpaper, it will store the short file name to allow automatic sound/paper switching to work).

Caption positions have been adjusted: left is replaced by middle and the right position takes into account the close icon.

The Windows Shutdown built-in command shows the Win95 shutdown dialog box allowing you to close windows, restart windows, or restart in MS DOS mode.

The Exit to Dos (Win95) built-in command will exit to MS DOS mode.

Right-drag and drop to the button bar can be used to configure a button (ie it is equivalent to left drag and drop with the Alt key held down).

You can use shortcuts (.LNK files) as commands but not as Icon files.

Coomands in Virtual Shell Folders (eg Network Dialers or their shortcuts) cannot be run directly from **Stiletto**. But you can access them by putting them on the Start Menu and sending the appropriate key sequence to the Task Bar. Or you can use the DunDial.exe program which comes in the **Stiletto** zip package.

You can use the built-in command Reset All Stiletto to restart all **Stiletto** bars if you dynamically change the display configuration, although this will usually be recognized automatically by **Stiletto**. If dynamic autosizing conflicts with the operation of your screen driver (eg for virtual desktops larger than the screen size), you can disable it by putting AutoDisplayChange=0 in the [general] section of you Stiletto.ini file.

The "Most Recent Cmds" built-in command cannot access document names; it can only access the associated .exe file.

If you position **Stiletto** on the task bar, use the Show **Stiletto** when Mouse Bumps Screen Edge feature to have the **Stiletto** bar appear on top of the task bar when the mouse bumps the screen edge nearest **Stiletto**.

Right clicking the desktop shows both the Windows 95 menu and any **Stiletto** menu you have associated with right clicking on the desktop. Because of the way Windows 95 operates, you must press and release the mouse, and you will get best results if, after the menu appears, you use left-clicking to access the menu items or dismiss the menu. Alternatively, you can access the **Stiletto** menu only by right-clicking the desktop if you Hide Program Manager.

Since Windows 95 already associates a menu with right-clicking the title bar, and since the right clicking to close option disables right-clicking on the task bar for the options menu, the option to right close a window by right-clicking the title bar does not operate in Win95. But if you set a menu to appear by right-clicking on the title bar, you can still see it by right-clicking on the window frame or menu bar. Or you can use the middle mouse key or shift-left instead.

If you use a caption position, and find that **Stiletto** does not always move to the active window, try using the ActiveRefresh internal option.

The appropriate routines are used for extracting icons.

Stiletto sound changing should work, but if you experience problems, you can disable sound writing with the NoSoundSave internal option.

You can disable Win 95 support with the with the Win95 internal option.

Alarm Log

You can ask **Stiletto** to log alarm events by using the Keep Alarm Log check box on the Alarm Setup dialog.

The log file will have the same name as the configuration file used in the instance of **Stiletto**, except that the file extension will be .alo. For example, the log file for the default configuration is stiletto.alo. The log is always placed in the same directory as the **Stiletto** ini file.

A log file entry will be written whenever an alarm rings. It will consist of the following fields, separated by blanks:

Current Year
Current Month
Current Hour
Current Minute
Current time expressed as seconds since 1970 01 01.
Alarm Year
Alarm Month
Alarm Hour
Alarm Minute
Alarm time expressed as seconds since 1970 01 01.
Alarm command and parameters.
Alarm work directory/message.

Icon Display Problems

Stiletto uses the "stretch" capability of your video driver to display icons as button labels. You will get an error message from **Stiletto** if your video driver reports that it does not have this capability.

Stiletto cannot display icons if your video driver does not support stretching.

The standard Microsoft Super VGA **Windows** video driver can be used to display icons.

In some cases, **Stiletto** may show the error message even though icons are displayed correctly. In this case, you can use the internal option CheckStretch in the ini file to turn off the error message.

Restarting a running instance of Stiletto

You can restart an instance of **Stiletto** with a new ini file command of the form:

```
c:\stilpath\stiletto.exe stilett2.ini restart stilnew.ini
```

where "stilett2.ini" is the file name of the configuration file used to start the instance of **Stiletto** you wish to restart and stilnew.ini is the new ini file name. It must be in the same directory as the original ini file. You must specify the file extension.

Stiletto File Commands

Stiletto has internal file manipulation commands for copying and deleting files. These are mainly intended for multiple commands where you need to make sure the file operation completes before the subsequent command.

To copy a file, use the builtin command Stiletto Exec and put copy FromPath ToPath in the command parameter field. Do not use long file names.

To delete a file, use the builtin command Stiletto Exec and put delete FilePath in the command parameter field. Do not use long file names.

Sending Key Strokes to Other Programs

You can send key strokes to other programs which are running or to programs which you launch with **Stiletto** using the Send Keys [builtin](#) command. (To send keys to programs when you launch them, you use *send as [shown](#) in the example below).

Use the command parameters to specify which window the keys are to be sent to and which keys to be sent. You must specify two parameters: the window to receive the keys and the keys themselves.

There are several ways to specify the window to receive the keys:

File Path	sends keys to program run from that "File Path"
*Title	sends keys to window with caption "Title"
PartTitle	sends keys to window with caption starting with "Part Title" (Note asterisk at end)
**PartTitle	sends keys to window with caption ending with "Part Title" (Note asterisks at start)
*	sends keys to current active window
**	sends keys to window of last command started by Stiletto
*t	sends keys to task bar. (see below for usage)

If the window file or caption that you want contains spaces, enclose it in double quotation marks.

To specify the keys to be sent, separate them by spaces and enclose them in double quotes. To specify an Alt-prefixed key, prefix it by a-; similarly use c- for Ctrl key, s- for Shift, and combine as needed (eg a-c- for both Ctrl and Alt). There are other special two-character sequences:

en	Enter key
sp	Space
qu	double quote
ta	Tab
es	Escape
au	Up arrow
ad	Down arrow
al	Left Arrow
ar	Right Arrow
fn	Function Key "n" (eg f1 for function key 1; do not actually use the letter n)

Example: "a-f n h e l l o f1" sends Alt-F, then "nhello", then function key 1.

If you have only one key to send, the surrounding double quotes are not needed. You can send at most 25 keys.

To start a program and send it keys at start up, use [multiple commands](#). For example, to start c:\ql\myprog and send alt-g n, specify

```
c:\ql\myprog.exe <*send ** "a-g n"
```

Note how ** is used to refer to the window of the program which **Stiletto** just started, namely myprog.exe.

Some programs show splash screens momentarily when they start; these will ignore keys sent to them. Use the wait command to wait for the splash screen to go away:

```
c:\ql\mysplash.exe params<*wait 5<*send "a-g n"
```

You can use *wait 0 to wait for about 1/3 of a second.

Your mouse and keyboard will "freeze" while keys are being sent; this is normal operation.

You can send keys to the task bar to activate items on the start menu; this is especially useful for activating non-programs such as network dialers which cannot be run as command lines from **Stiletto**. Put such a program on your start menu (eg by drag and dropping it from Explorer onto the Start Button). Suppose it is the third item from the top. Then the following parameters for the Stiletto Send Key command will activate the item:

```
*t "a-s ad ad ad en"
```

Explanation: Alt-s activates the Start Menu; the three down arrows go to the menu item, and the en Enter selects it. Or, if it is the only menu item starting with "i", then you could use

```
*t "a-s i en"
```

Minimizing a Window

Stiletto provides the capability to minimize an active window.

To use this capability, you need to put the built-in command "Minimize Window List" in a launch menu or as part of a sub-menu of a launch menu. When a launch menu is activated, a list of currently active windows is displayed. Selecting one of the windows causes the window to be minimized.

See also using a command to minimize.

Accessing a Launch Menu via Mouse Click

You can access launch menu 0 by clicking on the desktop or in a specified position of any window (or both). Use the menu setup dialog to specify how you want menu 0 to appear. If you have no launch menu named "0", then the menu which appears at the top of your list of menus in the menu contents dialog is used.

To make menu 0 appear by clicking on desktop:

Right:	right click on desktop
Middle	middle click
Right Center	click in middle 2/3 of screen avoids overlaps with Win95 menus

To make menu 0 appear by clicking on any window (including the desktop):

Title Right	right click in non-client area (title bar, window frame, menu, etc)
Title Middle	middle click on non-client area bar
All Shift-left	left click anywhere when shift key held down
All Shift-Middle	middle click anywhere when shift key held down
All Shift-Right	right click anywhere when shift key held down
All Middle	middle click anywhere
All Right	right click anywhere
All Center Right	menu appears with right click in middle 2/3 of screen

The non-client area refers to the window frame, the window title bar (not Win95), menus.

The built-in "Close/Min/Max/Hide Window Under Mouse" cursor commands are useful with these menus. For example, in Win3.1 you can simulate the Win95 title bar menu with these commands.

For Win 3.1x, if you set **Stiletto** to display launch menu 0 when you right-click on the desktop, you might still want to access another utility which attaches a menu to right clicking the desktop. For example, The Norton Desktop for Windows implements such a menu. Here is how to do it:

- Right click on the desktop to display the **Stiletto** menu.
- Move the mouse very slightly off the menu.
- Right click again to bring up the menu from the other utility.

Automatic Minimization of Inactive Windows

To help keep a neat desktop, you can use **Stiletto** to automatically minimize all windows except those belonging to the active task.

Since there are circumstances where you want to have more than one window open (eg when dragging and dropping), you control auto minimize mode with a command, not a configuration switch. You can turn auto minimize mode on or off by attaching the appropriate command to a button or launch menu.

To enable the command, you must first check the "Allow Auto Minimize" check box in the menu setup dialog. This box is automatically checked for active bars; you should control auto-minimizing from an active bar if you are using one as this will save system resources.

The AutoMinimize command can then be used to turn auto minimize mode off or on. Use the parameters edit box to set the function of the command:

on	to turn auto minimize on
off	to turn it off
toggle	to reverse the current auto minimize setting

To start **Stiletto** in auto minimize mode, put the auto minimize command on the startup menu.

You can use an active window list or the active bar or the Windows 95 task bar to switch among auto-minimized tasks.

It is possible to control which windows are automatically minimized by the Never AutoMin edit box on menu setup dialog. Put captions of windows you do not want minimized, separated by commas, in this edit box. If the entry in the edit box ends with a *, then windows with captions starting with the characters before the * will not be auto-minimized. For example

File Manager, Exploring*

prevents autominizing for both windows with the caption **File Manager** and windows with captions starting with **Exploring**.

It is also possible to turn off the autominimize process when specified windows become active; this is most useful for Help since you usually don't want the program which called help to be auto minimized. Put the captions of such windows in the Ignore For Automin edit box on menu setup dialog. If the entry in the edit box ends with a *, then windows with captions starting with the characters before the * will not cause auto-minimization. As a special case, if the entry consists of a ? alone, then no window with either the word Help or Search in its caption will cause autominimization. For example

Exploring*, ?

means that waindws with captions starting with **Exploring** and windows with **Help** in their captions will not cause autominimization.

Stiletto normally uses a system hook to control auto minimization; if this cause performance problems for you, can try a timer-based control with the ActiveRefresh internal option.

Automatically Hiding Windows

You can specify that **Stiletto** should automatically hide any windows, should they become visible and inactive.

To enable the command, you must first check the "Allow Auto Minimize" check box in the menu setup dialog. This box is automatically checked for active bars; you should control auto-hiding from an active bar if you are using one as this will save system resources.

Put the comma-separated captions of the windows you want to autohide in the Auto Hide edit box on menu setup dialog. If the entry in the edit box ends with a *, then any windows with captions starting with the characters before the * be hidden when inactive. For example,

HideMe

in this edit box will cause any windows with a caption of Hide Me to be hidden if inactive.

Stiletto normally uses a system hook to control auto hiding; if this cause performance problems for you, can try a timer-based control with the ActiveRefresh internal option.

New User Information

It appears this is the first time you have used **Stiletto**. Following is some basic information you may find helpful.

The default **Stiletto** button bar should be shown in the upper left of your screen. Move your mouse over it without clicking. A small yellow flyover help window will appear to show you which commands have been associated with each button. There are three commands per button: left mouse-click, middle (shift-left if you have a two-button mouse), and right.

Left click on the button marked Edit. Notepad should start. Right click on the same button and System Editor will start. Close both programs.

Left click on the button marked menu. A menu of commands will appear. Try selecting Program/Mgr shell and a list of your program manager groups will appear. Select Add Reminder Msg to create a timed reminder message

There are many ways you can customize **Stiletto**. For example, suppose you want to change the "edit" label to an icon.

Left click on the edit button and hold the mouse down without releasing until the configure button dialog appears. Notice the three sets of command entry controls for each of the left, middle, and right commands. At the top of the dialog, there is a drop box which says "edit". Use the drop down to change it to "icon from left". Press OK. The icon from Notepad (the left button command) will now appear on the button

There are more advanced configuration options. To see the full **Stiletto** configuration dialog, hold the Ctrl button down and left-click anywhere on the Stiletto button bar. You should see a tabbed dialog with these options:

Info shows Windows information and resource usage.

Buttons allows button commands, position, font, number to be set.

Menu/ Setup controls the format of the active window list, the mouse click used to start launch a menu on the desktop, and the format of menus.

Menu Contents allows you to change the contents of launch menus.

Alarm Setup controls low resource warnings and the sounds associated with alarms and chimes.

Alarm Details allows you to add or change alarms.

Sound controls the sounds played for windows and application events and allows you to specify how **Stiletto** should automatically change them.

Paper allows you to change the Windows wallpaper and to set up automatic re-wallpapering.

Key/Mouse allows you to assign commands to hot keys or to screen corners.

As an example: to move **Stiletto** to a new position, select the "Buttons" tab and click on the middle button at the top of the "Click Desired Stiletto Position" group. Press OK.

If you are interested in other Stiletto configuration options, see [hints and tips](#) and [new for this release](#). You can also click on the Contents button and use the >> buttons to browse through **Stiletto** help.

If you have any comments on **Stiletto**, you can contact me at bruce.switzer@canrem.com or bswitze@io.org.

