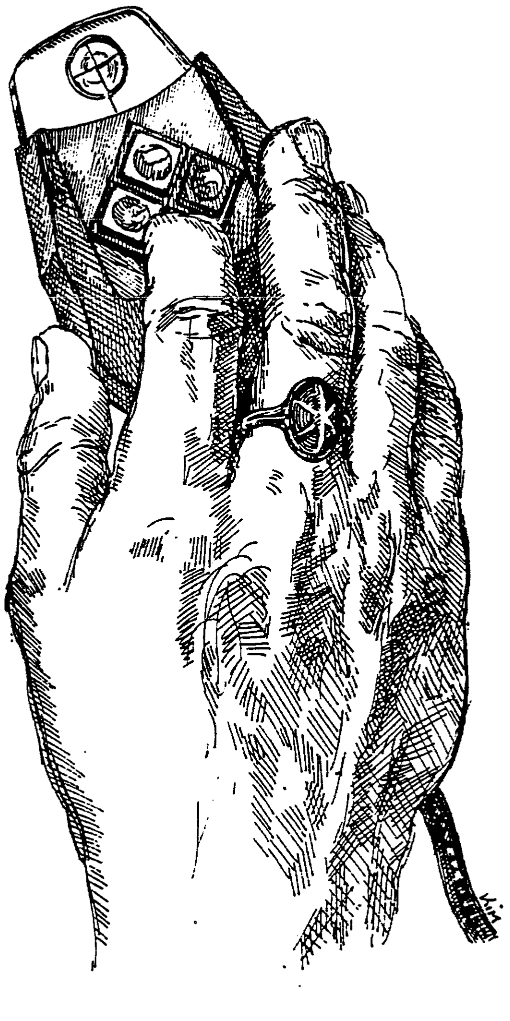
User's Guide to the Sapphire Window Manager



PERQ Systems Corporation
2600 Liberty Avenue
PO Box 2600
Pittsburgh, PA 15230
(412) 3550-0900

Keywords and index categories: <not specified>
Location of machine-readable file: [cfs]/usr/spice/spicedoc/aug84/intro/sapphire/winusr.mss

Accent and many of its subsystems and support programs were originally developed by the CMU Computer Science Department as part of its Spice Project.

The major part of the design and most of the implementation of the window manager was done by Brad Myers of PERQ Systems Corporation. The design grew out of his discussions with many people, including Gene Ball, the window designers at International Computers Limited, and various interested parties at CMU and PERQ Systems Corporation. Amy Butler and Dave Golub of PERQ Systems were instrumental in completing the window manager's implementation.

The information in this document is subject to change without notice and should not be construed as a commitment by PERQ Systems Corporation. The company assumes no responsibility for any errors that may appear in this document.

PERQ Systems Corporation will make every effort to keep customers apprised of all documentation changes as quickly as possible.

Table of Contents

1 Introduction

The window manager is the program that allows you to divide the screen into different areas, called windows, and to monitor and control the processes running in the windows. This document describes from a user's point of view how the window manager operates with the Shell. Section 10 of this document is a glossary of terms relevant to the window manager, and the last page of this document is a summary of the commands.

The details of window management as used within a particular application program will be covered in the document for any such program. A separate document ("The Window Manager" in the *Accent Programming Manual*) describes how programmers can incorporate the window manager into application programs.

1.1 Key Concepts

For readers who are not familiar with window managers, this section gives background information about window managers in general and the Accent window manager in particular.

1.1.1 Windows

If one were working on a number of different pieces of paper at a desk, one would typically separate the pages physically by having them at different places on the desk. Similarly, a window manager separates a number of different computer tasks by assiging each a separate area of the computer screen. Each area of the screen is called a window. The PERQ, like most personal work stations that support multiple processes, allows the user to assign the different processes to different windows in order to keep track of them separately.

In general, the window manager supports what is known as the "Covered Window Paradigm," a method for presenting multiple windows on the screen at the same time and allowing the various windows to overlap (see Figure 1). The windows can be thought of as pieces of paper. Windows can be moved around on the screen, just as papers can be moved around on a desk. One window can be brought on top of another, thereby partially or wholly covering that window. A window that cannot be entirely seen because other windows are on top of it is said to be covered. A window which is covered by all other windows occupying the same area of the screen is on the bottom. A window can extend off-screen in any direction (somewhat like papers hanging off the edge of a desk). The graphics and text in one window do not affect the graphics and text in other windows. A program can be running in a window even if it is partially or wholly covered by other windows or if it is off-screen.

PERQs are equipped with a tablet and a three-button or four-button mouse. When the mouse is moved around on the tablet, the computer echoes the mouse's position by displaying a cursor on the screen which follows the mouse movement. Commands can be given to the window manager in three ways: with the mouse, from a command menu, or from the keyboard.

1.1.2 Listener

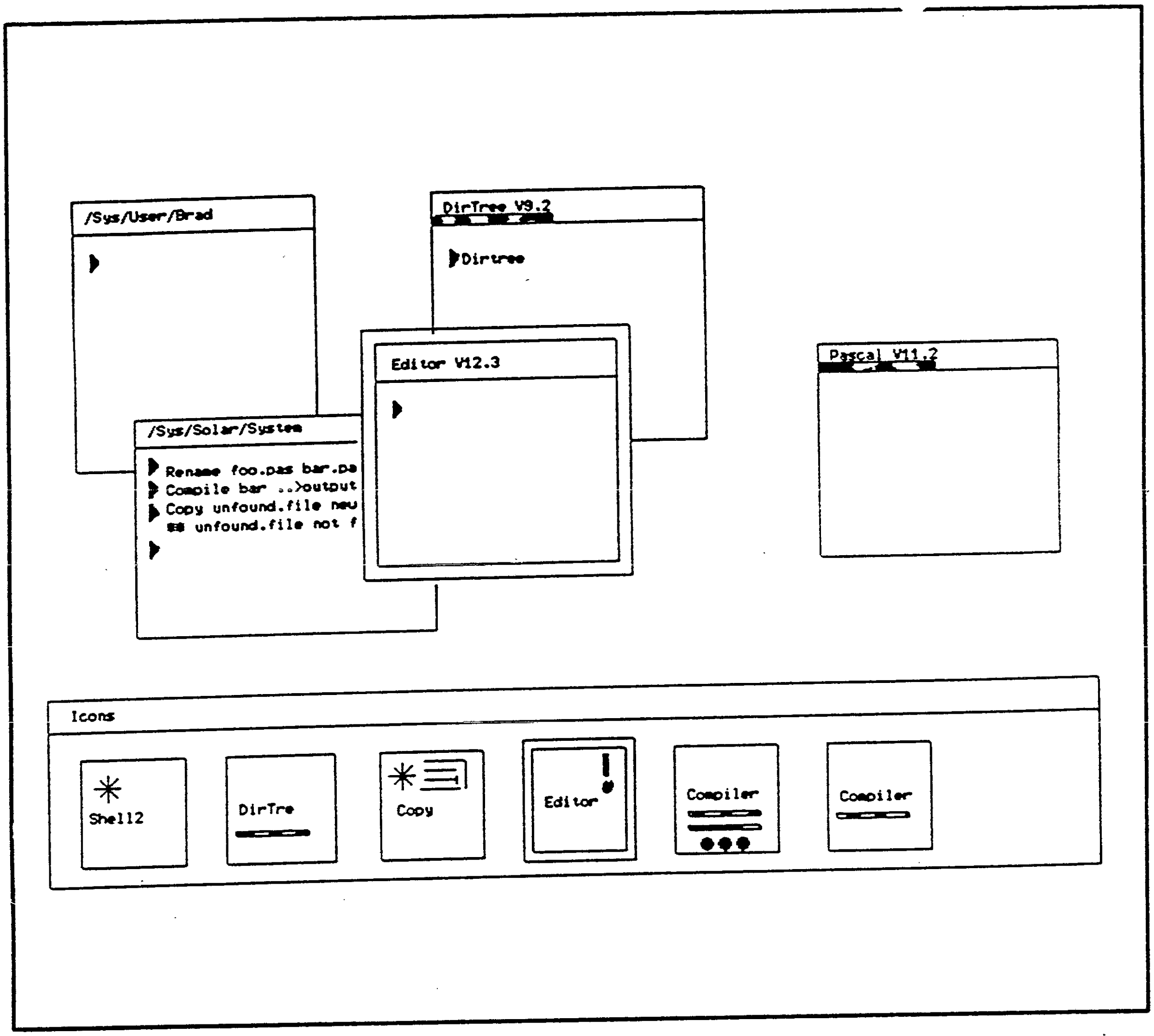
While there is only one keyboard and mouse on a machine, the screen can contain many windows. Therefore, one window has to be identified as the window to which keyboard operations are currently directed. This window is called the Listener because it is the window that is currently listening to the keyboard.

1.1.3 Icons

To allow you to monitor and control the programs that are running in windows (uncovered, covered, or off-screen), the window manager provides icons (small pictures which give information about each window, regardless of its position). The icons tell you whether a program is still running, is waiting for input from you, or is finished, and whether the window is off-screen. All of the window manager's icons are collected together in one window, the Icon Window (usually at the bottom of a portrait screen or at the side of a landscape screen), so that all of the icons can be manipulated as a group (for example, if you want to cover them with another window or move them off-screen).

The Process Manager Window (which the system places at the top of the screen when you enter Accent) and the Icon Window can be treated like any other window.

Figure 1 shows five user's windows plus the Icon Window. In the title line of each window is arbitrary text set by the program being run in each. The first window is in the Shell and the previous program run there exited with an error (the icon Shell2 with the bug). DirTree is executing in the second window and is showing random progress in the title line (below the text) and in the icon. The third window, like the first window, is in the Shell, and the last program run there also exited with an error. This shell is waiting for user input (the keyboard). The Editor window has a gray border to show that it is currently the Listener. Its icon is similarly marked. It is trying to get the user's attention by using the exclamation point picture in the icon. The next window is off-screen (shown by the three dots at the bottom of the icon). The compiler is running in this window. The top progress bar shows random progress and the bottom progress bar shows that it is part of a command file that is about 25% done. The final window is also compiling. It is working on a program which starts with "WinM" and random progress is shown in both the icon and the title line for the window.



```
Dirtree
```

```
Rename foo.pas bar.pa
Compile bar ..>output
Copy unfound.file neu
** unfound.file not f
```

Figure 1. Sample Windows

1.1.4 Cursors

The window manager uses a number of cursors, each of which is associated with a function. The cursor pictures and their functions are shown in Figure 2. When you give the window manager a command, the cursor changes to the appropriate picture or to the window manager default cursor. If you are issuing a command using the mouse, the cursor picture appears when you press the button and stays until you release the button. When you release the button, the command is executed. The cursor pictures indicate their functions so that you can tell at a glance whether you have given the correct command. If you see that you have given the wrong command, just move the mouse until you see the "Cancel" cursor before you release the button.

When the user presses down on a mouse button while the cursor is in the title line, border, or icon of a window, the cursor will change to one of the window manager's cursors to indicate what will happen next. The command is executed when the button is released. If the operation requires multiple presses (for example, growing a window), the cursor picture will change at each step to show what is expected next. The cursors are shown on the following page.

1. The Accent system default cursor.
2. The window manager default cursor(waiting for input).
3. Make the window in which the cursor is located the Listener.
4. Bring the window to the top.
5. Send the window to the bottom.
6. From the icon, make the window on-screen, on-top, and Listener.
7. Get a popup menu from a window.
8. Make the window smaller or take it off-screen (the < command).
9. Bring the window on-screen (icon only) or make it full-screen (the > command).
10. Pick a corner or side from which to move or resize the window.
11. Resize the window.
12. Move the window.
13. Cancel command.
14. Specify the top left corner of the window.
15. Specify the top side of the window.
16. Specify the top right corner of the window.
17. Specify the right side of the window.
18. Specify the bottom right corner of the window.
19. Specify the bottom side of the window.
20. Specify the bottom left corner of the window.
21. Specify the left side of the window.
22. Identify the window for this icon.
23. Get a popup menu from an icon.

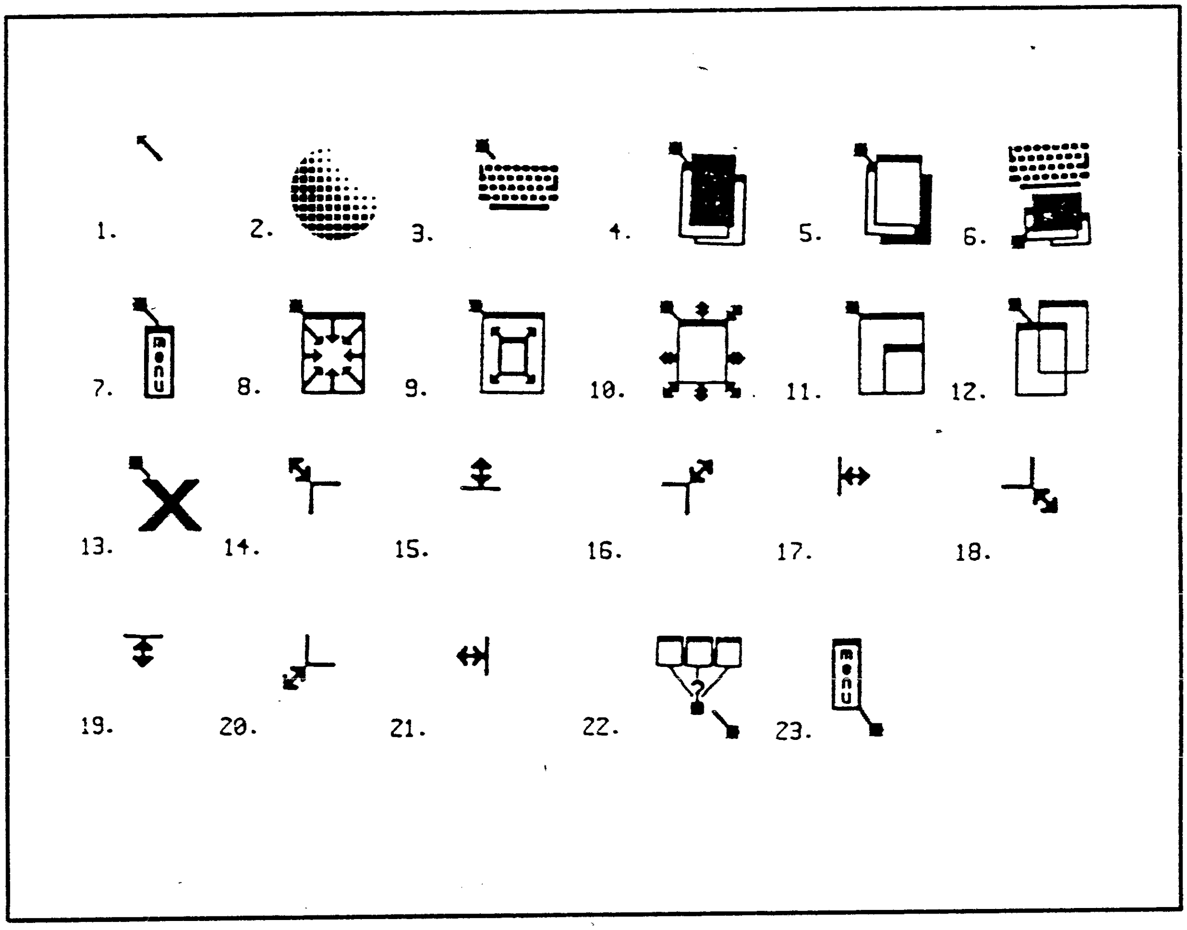


Figure 2. The Window Manager Cursors

1.2 Window Features

Each window is rectangular and may extend off the screen in any direction. A window can be moved; changed in size; brought to the top or sent to the bottom of the stack of windows occupying the same area; made full screen; sent off-screen; or restored to its original position. For each window its title line, borders, and icon are special areas. The window manager commands are given with the mouse by positioning the arrow cursor (cursor #1 in Figure 2) in one of those areas, as explained in detail in Sections 2 through 10.

Figure 1 shows a screen containing several windows. The following sections give more information about title lines, borders, progress bars, and icons.

An application program may give windows a number of special properties. For example, the windows may not have borders, title lines, or icons, they can be restricted from moving or growing, they can be transparent so that the picture underneath shows through, and they can remember the picture underneath so that the area reappears later (as happens with popup menus in the Shell).

1.2.1 Title line and borders

The title line, located across the top of each window, is black with white letters. It serves three functions. First, it displays a line of arbitrary text specified by the Shell or by an application. Second, it contains a progress bar, explained in the next section. Third, it can be used in conjunction with the mouse to perform operations on the window (by locating the cursor within a certain portion of the title line and pressing a mouse button, as explained in Section 1.3.1).

The border goes entirely around the window (outside of the title line). The Listener window is identified with a special border around the outside of the window (a gray area between two hairlines), both in the window itself and in its icon.

1.2.2 Progress bars

A progress bar in the title line of each window shows how much of an operation has been completed. The progress bar appears on a line below the text string and fills in (in reverse video) from left to right as the process runs. In some cases the application cannot determine how complete an operation is, but nevertheless it is helpful to the user to show "random progress" (an indication that the operation is still being processed). In such cases random progress is shown by displaying blinking lines in the progress bar.

The progress bar is repeated in the icon so that the user can determine the progress of operations in windows that are off-screen or covered. In the icon only, there is a second progress bar that is used for command files so that you can tell how much of a command file has been processed.

1.2.3 Icons

An icon is a small picture that represents a window (see Figure 1). Icons contain information about the status of the corresponding window and about the process running in the window. They permit the user to easily monitor and control multiple processes and windows.

All the icons are grouped together in one window, the Icon Window. The title line of the Icon Window can be used to issue commands to control the icons as a group. (Button presses inside the title line of the Icon Window mean the same thing as for any other window.) Button presses inside an icon, however, are interpreted as an operation on the window associated with that icon.

The Icon Window can be covered or moved off-screen like any other window. There is no icon for the Icon Window itself.

The order of icons within the Icon Window is random. A new icon goes in the first available space. When a window is deleted, the associated icon goes away but the rest of the icons do not move. The user can compact the icons to remove these spaces if desired (see section 9.3).

The user is free to move or grow the Icon Window. The icons will be automatically organized for the new size or shape. The default position for the icons is one row across the bottom of the screen for Portrait screens (which hold 11 icons) and down the right side of the screen for Landscape screens (which hold 15 icons). If the Icon Window is full and another icon is needed, the Icon Window automatically expands along its shortest dimension and moves so that all the icons are on the screen.

Each icon is divided into a number of sections. The top of the icon can contain three pictures, used to show the state of the process running in the window. On the left will be a "bug" when an error has been reported in the window. When there is no error, the bug will be absent. In the center is a "keyboard" when the window is waiting for input from the user. On the right is an exclamation point which is for application's use. For example, a Mail program might bring up the exclamation point (called the "attention" picture) when new mail comes in.

Below the pictures is a six-letter text string used to identify the window. This will usually be the first six letters of the text name of the program being run in the window. The name that appears in the icon is always guaranteed to be unique with respect to all other icons. If an application specifies a name that conflicts with another window's name, a number is appended to the end. If the name is already six characters long, the last character of the name is changed to a number. Thus, if "foo" exists, another foo would become "foo2". If "Pascal" existed, another Pascal would become "Pasca2". The Shell will set the name to the program being run, but some programs may explicitly set it to some other name.

Below the name are two progress bars. The first bar operates like the progress bar in the title line of the window itself. The second bar shows progress in command files. Below the progress bars is an area that contains three big dots (an ellipsis) if the window has been sent off-screen.

The border of each icon changes the same way as the associated window's border to show whether the window is the Listener or not. Thus it will be either a hairline if not the listener or two hairlines surrounding a gray area if it is.

1.3 Commands - General Information

There are three ways to issue commands: with the mouse, with the command menu, and with the keyboard. For most operations you have a choice between at least two ofthese methods.

The following sections give some general instructions for issuing commands in each of these three ways. The specific commands are given in sections 2 through 10, organized by function, and they are summarized on the last page of this document.

If you receive an error message from the window manager, please report it to your system administrator or PERQ Systems. The error messages are listed in the document "The Window Manager" in the *Accent Programming Manual.*

1.3.1 Mouse commands

Using the mouse to issue commands to the window manager involves three steps:

1. positioning the cursor in the appropriate area,
2. pressing the appropriate button (after which a cursor appears, depicting the action you've chosen), and
3. releasing the button (after which the action is completed).

The window manager is written for a three-button mouse. If you have a four-button mouse, the white button corresponds to the left button, the yellow to the middle button, and the green to the right button; the blue button serves no function.

A cursor in the form of an arrow echoes on the screen the position of the mouse on the bitpad. To invoke the window manager commands using the mouse, you must first position the arrow cursor on either the title line or the icon of the window you wish to affect (it does not have to be the Listener). Be careful not to press a button within the Listener, as all button presses within the Listener are interpreted by the application program running in that window.

The title line of each window is divided into three sections: left, middle, and right. Since often only one side of a window is visible, the left and right sections of the title line serve the same functions. Using the three buttons on the mouse and the two areas of the title line, there are six functions that can be performed from the title line (discussed in detail in the section shown below):

- send to top of stack of windows (Section 6.1)
- send to bottom of stack of windows (Section 6.1)
- show the command menu (Section 5)
- switch from original size to full-screen (Section 6.3)
- switch from full-screen to original size to off-screen (Section 6.4)
- change position or size (Section 6.5)

The cursor positions and the buttons for these commands are as follows:

```
     Title Line                  Title Line
      Left or                      Middle
     Right End
 -----------------   ----------------------------------
|     BUTTONS     | |             BUTTONS              |
| Top |Bottom|    | |Orig. size |Full-screen  |Change  |
|  of | of   |Menu| |   to      |to original  |position|
|stack|stack |    | |full-screen|to off-screen|or size |
 -----------------   ----------------------------------
                                                      |
                                        Move--black blocks/
                                              any button
                                        Grow--white blocks or
                                           corners/any button
```

After you position the arrow cursor in the desired area and press the appropriate button for the command you wish to give, the cursor will be changed to a picture that represents the action that will be completed when the button is released. Figure 2 shows the cursors used in the window manager. Release the button to complete the action.

When you are issuing a command to the window manager, you cannot press a second button before you release the first button. This is because the window manager treats the pressing and releasing of a button as one mouse event–pressing brings up the appropriate cursor and releasing executes the command. If the first button pressed is not the same as the last button released, the command is ignored. (This may differ in applications, because presses and releases are treated as separate events in applications.)

If after you press a button and obtain a cursor you realize that you have picked the wrong area or wrong button, you can correct the action before you release the button. Move the cursor across the title line to switch to another function (the cursor picture will change) or move the cursor outside the title line to abort the action.

Pressing a button within an icon affects the window represented by that icon. There are three functions that can be performed from an icon:

- make the window on-screen, on-top, and Listener in one operation
- identify the window represented by the icon
- show the command menu

The buttons used to issue these commands are as follows:

BUTTONS		
On-screen, on-top, & Listener	Identify window	Menu

1.3.2 Menu Commands

The command menu contains nearly all of the commands (there are a few commands that can be issued only from the keyboard). Section 5 explains how to obtain and use a menu.

You can request a menu from any window or its icon, whether or not it's the Listener. If the command applies to one regular window only (rather than the Icon Window or all windows), the command will be executed in the window from which you request the menu.

1.3.3 Keyboard Commands

All operations that can be performed with the mouse or from the menu, plus a few others, can be performed with keyboard commands. Keyboard commands affect only the Listener (except for a few commands that affect only the Icon Window or that affect all windows).

All keyboard commands must be preceded with a prefix of CTRL-DEL on the PERQ and with CTRL-DEL or SETUP on the PERQ2. (To use CTRL-DEL press and hold the CTRL key while you press the DEL key.) Release the prefix key before you type the command. When the window manager prefix key is pressed, the window manager suspends any process that is running in the Listener and changes the cursor shape to a window manager cursor. The next character that is typed will be interpreted as a window manager command. The commands can be typed in lower or upper case. Typing another character returns control to the process in the Listener; therefore if you want to give more than one command you must repeat the prefix (except for the + and - commands explained in section 3).

If some process has the mouse trapped or turned off or otherwise disabled, you can resume giving mouse commands by first giving a keyboard command (which will release the mouse).

2 Creating Windows

When you enter Accent, a window is created automatically for the Process Manager. Another window is created which runs your initial shell command file.

Additional windows can be opened in one of the following ways:

1. type *<prefix>s*; or
2. select "shell" on the command menu.

After you do either of the above, you must indicate the upper left and lower right corner as follows. After the appearance of a blinking cursor in the form of an upper left corner with an arrow (cursor #14 in Figure 2), position the cursor where you want the upper left corner of the window and press and release any button. The cursor will change to a lower right corner with an arrow (cursor #18). Move the cursor to where you want the lower right corner (a hairline box showing the outline of the window will follow the cursor) and press and release any button.

The window manager will not permit you to create a window that is smaller than a minimum size (which is very small) specified by the window manager.

If after you open a window you find that you need to make it bigger, obtain the menu (from the title line or the icon) and select the reshape command explained in Section 6.6.

After you create a window, you cannot use the window until you make it the Listener.

3 Designating the Listener

As explained in the Introduction, the Listener is the window that is receiving commands from the keyboard. The window that is currently the Listener has a narrow gray band around the border.

There are several ways to designate a window as the current Listener:

1. Move the cursor into the desired window and press any button. A picture of a keyboard (cursor #3 in Figure 2) appears. When the button is released, the border changes to show that the newly selected window is now the Listener. (Moving the cursor without pressing does not change the Listener, nor does moving the cursor outside a window while pressing.)
2. Either select "listener menu" on the command menu or type *<prefix>l*. After either action, a menu appears which lists all of the windows; select the window you wish to make the Listener.
3. To bring a window back if it is off-screen, bring it to the top if it is covered, and make it the Listener all in one operation, press the left button while in the icon (cursor #6 in Figure 2 appears).
4. Type *<prefix>+* or *<prefix>-* to move around a ring of windows. (The window manager remembers which windows have been the Listener in what order and keeps their names in a circular buffer.) To go back to the window that was the most recent Listener before the current Listener, move back by typing *<prefix>-*. To go to the window that was the least recent Listener (or the most recently created window that has never been the Listener), move forward by typing *<prefix>+*.

These two characters, - and +, are easy to remember in conjunction with these functions, but either the upper- or lower-case characters of each key will work (← or - to move back and + or = to go forward).

Whether moving forward or backward the window manager default cursor (#2 in Figure 2) appears. The ring is re-arranged each time you change Listener. You can give multiple +'s and -'s without repeating the prefix key. After you have used the + or - command, you can leave the window manager and return to an application program by typing *<prefix>x* (for "exit window manager" or just by typing *q*. If you want to go back to the window that was Listener before you gave the + or - command and resume the application program, type any illegal command such as the BACKSPACE key.

4 Obtaining Help

On-line help for the window manager can be obtained in any of the following ways:

1. select "help" on the command menu;
2. press the <prefix>HELP keys; or

3. type *<prefix>h.*

A small window in reverse video appears in the center of the screen. It contains a summary of the mouse and keyboard commands. To remove the help window and restore the portion of the screen that it covers, type any character.

5 Obtaining the Command Menu

The command menu can be obtained in any of the following ways:

1. press the right button while on the left or right section of the title line (cursor #7 in Figure 2 appears);
2. press the right button while in the icon (cursor #23 in Figure 2 appears); or
3. type *<prefix>?* (Listener only).

When you release the button, the menu appears. Commands on the menu can then be selected by moving the cursor to the desired command and pressing and releasing any button. The command is executed on the window from which the menu was requested. To abort the menu without making a selection, press outside the menu area.

Nearly all of the commands that are available in the window manager are listed on the menu (see the summary on the last page of this document).

6 Moving and Changing the Size of Windows

6.1 Top or Bottom of Stack

If a window is on top of the stack, it is not covered by any other windows. If a window is on bottom, it is covered partially or wholly by any other windows that occupy the same area of the screen.

A window can be brought to the top or sent to the bottom in any of the following ways:

1. position the cursor in the left or right section of the title line. Press and release the left button to bring the window to the top or the middle button to send it to the bottom (cursor #4 or #5 appears);
2. on the command menu select either "top" or "bottom,"

   or
3. type either *<prefix>t* for "top" or *<prefix>b* for "bottom" (Listener only).

If a screen is on the bottom and is not accessible, you can move it to the top by requesting the command menu from its icon.

Sending a window to the bottom is useful for finding the other windows that are covered by that window.

6.2 On-screen, On-top, and Listener

There is a special command which can be used to bring a window on-screen, put it on top of the stack, and make it the Listener in one operation. Press and release the left button while in the icon of the desired window (cursor #6 appears).

6.3 From Off-screen to Original to Full-screen

There is a special command which takes a window up through a hierarchy, from off-screen to full-screen. If a window is off-screen when you issue the command, the window will be returned to its original position on the screen. If the window is on-screen, it is made full-screen. (Thus, to bring a window from off-screen to full-screen requires issuing the command twice.) This operation can be performed in any of the following ways:

1. if the window is on-screen and you want to make it full-screen, press and release the left button while on the middle section of the title line (cursor #9 appears); or
2. type *<prefix>* > (Listener only);
3. if the window is off-screen, from its icon select ">Bigger" on the command menu.

When a window is made full-screen, its old position is remembered so that it can be returned to its original position in a single "go back" operation (explained in section 6.4 below). In the Shell, "full-screen" uses the entire window outside the Icon Window, but maintains the Icon Window. This is useful for making the window for the current process (for example, the editor) large while monitoring other processes through their icons. (Application programs may define "full-screen" differently.)

The size of the window is saved each time the full-screen operation is executed. Therefore if you specify full-screen and then change the size of the window, the "go back" command will return the window to the original size (before the full-screen command). However, if you specify full-screen, change size, then specify full-screen again, the intermediate size will be used for the "go back" command.

6.4 From Full-screen to Original to Off-screen

A special "go back" command takes a window down through a hierarchy from full-screen to off-screen. If the window is full-screen when you issue the command, the window is returned to its original size (see section 6.3 for a discussion of what size is considered to be the "original" size). If the window is not full-screen, it is sent off-screen. This operation can be performed in any of the following ways:

1. press and release the middle button while on the middle section of the title line (cursor #8 appears);
2. select "<Smaller" on the command menu; or
3. type *<prefix>* < (Listener only).

Three dots in the icon indicate that the window is off-screen.

Sending a window off-screen is useful for two reasons. First, it allows the screen to be less cluttered since off-screen windows will not show up when other windows are moved around or sent to the bottom. Second, sending a window off-screen makes the refresh calculations for all other windows more efficient.

A window that has been taken off-screen can be brought back to its original on-screen position by using its icon to request the command menu and choosing ">Bigger."

You should use one of the methods shown above to take a window off-screen rather than just moving it off-screen with the move command discussed in the next section. Otherwise, the window is included in the refresh operation even if it's off-screen and its on-screen position is not remembered.

6.5 Position or Size

Windows can be moved or "grown" (increased or decreased in size) in any direction. To perform either operation, perform the sequence described below.

1. Indicate in one of the following ways that you want to move or grow the screen:
2. a. press and release the right button while on the middle section of the title line (cursor #10, size/move, appears);
   b. select "move window" or "grow window" on the command menu; or
   c. type *<prefix>m* for "move" or *<prefix>g* for "grow" (Listener only).
3. Indicate the direction in which you wish to move or grow the window by selecting a control point as follows:
4. a. To move a window, position the cursor precisely on one of the black squares on the window's border and press any button. Cursor #12 appears and stays until you release the button; then a different cursor (#14 through #21) appears.
   b. To grow a window, position the cursor precisely on the white square on the side or corner in the direction you wish to increase or decrease and press any button. Cursor #11 appears and stays until you release the button; then a different cursor (#14 through #21) appears.
5. Move the cursor to the desired location. As the cursor is moved, a hairline box is drawn to show the bounds of the window.

6. When the window reaches the desired location or size, press and release any button.

If you have trouble at first remembering which blocks to use, remember to look at the cursor which appears when you press a button. The command is not executed until you release the button, so if you picked the wrong block move the cursor before you release the button.

To abort the procedure at any time, press any character on the keyboard.

Windows can be moved partially off-screen in any direction if you wish.

As an alternative to Move or Grow you may wish to use Reshape, explained in section 6.6 below.

6.6 Reshape

Reshape is a useful alternative to Move or Grow, as it allows you to specify entirely new corners for a window. To use Reshape, do the following:

1. Either select "reshape window" on the command menu or (Listener only) type *<prefix>r.* Cursor #14 appears.
2. Position the cursor at the new location for the upper left corner and press and release any button. Cursor #18 appears.
3. Repeat step 2 for the lower right corner. (As you move the cursor, a hairline box will show the new window size.)

To abort the process, press DEL; the original window will return.

7 Debugging or Stopping Processes

7.1 Debugging

To send a process into the debugger, select "debug process" on the command menu or (Listener only) type *<prefix>d.* You will be asked to open a window for the debugger.

7.2 Suspending a Process

To suspend a process, select "suspend process" on the command menu or (Listener only) type *<prefix>z.*

7.3 Resuming a Process

To resume a process that you had suspended, select "resume process" on the command menu or (Listener only) type *<prefix>q* (for "quit window manager").

7.4 Cancelling a Process

To cancel (abort) a process, select "cancel process" on the command menu or (Listener only) type *<prefix>c.*

7.5 Killing a Process

Killing a process aborts the process as well as any command files, and the Shell returns to the top level, waiting for user-input. To kill a process, select "kill process" on the command menu or (Listener only) type *<prefix>k.*

If you wish to also delete the window, see section 8.

7.6 Quitting the Window Manager and Returning to the Application

To leave the window manager and go back to the application program, type *<prefix>x* (for "exit window manager"). This command should be given after moving in the Listener ring with the + or - commands.

8 Deleting Windows

To kill a process and delete the window in which it was running, select "eradicate window" on the command menu or (Listener only) type *<prefix>e.* This kills all processes running in that window, including the Shell.

9 Working with Icons

9.1 Bringing the Icon Window On-screen, On-top

To bring the Icon Window back if it is off-screen and to bring it to the top of the stack in one operation, select "icon window" on the command menu or type *<prefix>i.* The default window manager cursor (#2) appears. You can execute this command from any window.

9.2 Identifying the Window Represented by an Icon

To identify which window goes with an icon or vice versa, do one of the following:

1. from the icon, press and release the middle button (cursor #22 appears);
2. from the window, select "find icon" on the menu; or
3. from the window (Listener only), type *<prefix>f.*

The icon and its matching window are video-inverted, and lines are drawn from the four corners of the icon to the four corners of the window. The lines are blinked on and off a number of times. If the window is covered, the screen area where it would be if it were on top is shown. If the window has been sent off-screen using the off-screen command (see section 6.4), then the icon simply blinks and no lines are drawn. If the window has been moved partially off-screen, however, lines are drawn toward the ends of the window.

9.3 Reorganizing the Icons

When a window is deleted, its icon is erased but none of the other icons move. To reorganize (compact) the Icon Window to remove the blank spots and to redraw the window, select "organize icons" on the command menu or type *<prefix>o*. The default window manager cursor (#2) appears. This command can be executed from any window.

9.4 Monitoring the Icons

The following symbols are used in the icon to tell you what is happening in the corresponding window:

keyboard	program is waiting for user input
bug	error encountered
exclamation point	window needs attention (symbol may be used differently in application programs)
6-letter text string	window name (a unique identification)
progress bar 1	progress in program (proportion completed or random progress)
progress bar 2	progress in command file (proportion completed or random progress)
3 dots	window is off-screen

10 Glossary

The following definitions are given for your information.

Accent: a multi-tasked, message-based operating system for the PERQ. The window manager described in this document runs only under Accent.

Application: any program that uses the window manager.

Application Interface: the set of messages or procedures and exceptions that are provided to the application by the window manager. The window manager allows applications to display text and graphics in windows and to create, manipulate and delete windows. The interface is not apparent to the user.

Covered Window Paradigm: a method for presenting multiple windows on the screen at the same time and allowing the various windows to overlap. When two windows overlap, one is said to be "on top" when it can be seen in its entirety. The other window is said to be "covered" since it is at least partially obscured by the first window. With more than two overlapping windows, there is an ordering of windows from "top" to "bottom" where the top one is fully uncovered, and the bottom one is "underneath" all other windows.

Cursor: a picture that follows a pointing device. In the window manager the cursors are pictures that indicate the action the user has chosen.

Icon: a small picture that represents a window. Each icon will be associated with a particular window and will contain information about the status of that window and the process running in it.

Listener: the process that is currently receiving keyboard input. In a multi-process multi-window system, the keyboard needs to be multiplexed over the processes and windows. Only one process at a time may receive the keyboard characters. This is the Listener process. "Listener" is also used to refer to the window containing the Listener process.

Mouse: a device which the user moves on a tablet in order to move a cursor on the screen. It may have one, three or four buttons (used to issue commands), with three being the most common number. The user interface to the window manager uses only three buttons. These map to the four colored buttons on a Summagraphic mouse as follows: Left = White, Middle = Yellow, and Green = Right. The user interface to the window manager does not use the blue button; however, application programs using the window manager may use it. (A mouse, especially if it has four buttons, is often referred to as a puck.)

Output: as used in this document, all graphical and textual output to the screen. Included are displays of text, line drawings, pictures, and characters echoed when typing.

Shell: a program or process that is in charge of running other programs. The shell takes requests from a user or a command file and executes the programs represented by those requests.

Tablet: a device attached to a computer and used with a mouse to move a cursor on the screen (therefore a tablet is referred to as a pointing device).

Typescript: normal text input to the system from the keyboard. The typescript is linear and can be thought of as a "glass teletype" or the typing that would appear on a printing terminal. Thus, it excludes typing to screen editors and other application programs that interpret individual characters as commands. Most user typing will be handled by the typescript package.

User: a person who is sitting in front of the PERQ typing on the keyboard and using the mouse.

User Interface: the part of the window manager that the user sees and uses directly. This includes the pictures displayed on the screen, the operations the user performs (using the keyboard and mouse) to affect the windows, and the responses of the window manager to those actions.

Window: any rectangular area of the screen that is controlled by the window manager. It may extend off the screen in any direction, or may even be off the screen entirely. Windows are clipped to the screen's rectangle (that is, the visible portion is cut off at the boundary).

Window Manager: a component of the Accent operating system that allows users and programs to create, modify and output text and graphics to windows. It may also be in charge of collecting and distributing input from the user to the appropriate process. The Window Manager (at the user's request) changes which windows are covered and uncovered by moving the windows to the top and bottom.

SUMMARY OF WINDOW MANAGER COMMANDS

Mouse Commands

```
     Title Line                 Title Line                  Title Line
      Left End                    Middle                     Right End
   ----------------   --------------------------------------
  |     BUTTONS    | |              BUTTONS                 |
  | Top |Bottom|   | |Original size |Full-screen  |Change   |   Repeat of
  | of  | of   |Menu| |     to       |to original  |position|   left end
  |stack|stack |   | | full-screen  |to off-screen|or size |
   ----------------   --------------------------------------
                                                      |
            --------------------------       Move--black blocks/
           |          BUTTONS         |            any button
  Icon:    |On-screen,|         |     |      Grow--white blocks or
           |on-top, & |Identify| Menu |            corners/any button
           |Listener  | window |      |
            --------------------------
```

* For the following keyboard commands, prefix is CTRL-DEL on the PERQ and CTRL-DEL or SETUP on the PERQ2.

Menu Commands	Keyboard Commands*
Shell (create new window)	<prefix> s
Eradicate window (kill process and delete window)	<prefix> e
Help	HELP or <prefix> h
Listener menu (to designate new Listener)	<prefix> l
Bottom (send to bottom of stack)	<prefix> b
Top (send to top of stack)	<prefix> t
>Bigger (from off-screen to original to full-screen)	<prefix> >
<Smaller (from full-screen to original to off-screen)	<prefix> <
Grow window (using corners or white blocks)	<prefix> g
Move window (using black blocks)	<prefix> m
Reshape window (redefine upper left & lower right corners)	<prefix> r
Icon window (bring on-screen, on-top)	<prefix> i
Organize icons (compact the icons)	<prefix> o
Find icon (identify corresponding window)	<prefix> f
Debug process (send window to debugger)	<prefix> d
Cancel process	<prefix> c
Kill process (including command files)	<prefix> k
Suspend process	<prefix> z
Resume process	<prefix> q

Keyboard Commands Not Available from Menu	
Obtain menu	<prefix> ?
Forward in ring	<prefix> +
Backward in ring	<prefix> -
Exit window manager and return to application	<prefix> x