
x6530 User's Guide

Product Version x6530 C32

Release ID D10.00/C30.09

Edition Print Date January 1993

Part Number 082628

Abstract This manual explains how to use and customize the x6530 Emulator, an X client that gives you access to Tandem systems from a variety of hosts in an X Window System environment.

Document History	Edition	Part Number	Product Version	Release ID	Print Date
	First	026215	x6530 C30	C30	October 1990
	Second	082628	x6530 C32	D10.00/C30.09	January 1993

New editions incorporate any updates issued since the previous edition.

Release ID Note Release ID specifies the release in which the software functions described in an edition are first available on a standard SUT (site update tape).

Copyright Copyright © 1992 by Tandem Computers Incorporated. All rights reserved. No part of this document may be reproduced in any form, including photocopying or translation to another language, without the prior written consent of Tandem Computers Incorporated. Printed in the United States of America.

Export Statement Export of the information contained in this manual may require authorization from the U. S. Department of Commerce.

Examples Examples and sample programs are for illustration only and may not be suited for your particular purpose. Tandem does not warrant, guarantee, or make any representations regarding the use or the results of the use of any examples or sample programs in any documentation. You should verify the applicability of any example or sample program before placing the software into productive use.

Ordering Information If you would like manual ordering information, contact the following:

- Domestic U.S. Customers—Call 1-800-243-6886.
- International Customers—Contact your local sales representative.

Please Comment If you have questions or problems concerning the content of this document, please use the Reader Comment Card located in the back.

New and Changed Information

This is the second edition of the *x6530 User's Guide*. This revision incorporates the following changes to the manual:

Section 1: Descriptions of the user interface are changed to reflect the migration of x6530 from the Athena standard to the Motif standard:

- Changes in screen appearance
- Changes in mouse usage
- Changes in scrollbar appearance and use
- Addition of cascading menus for configuration displays and custom commands

Also included is a new description of the WINDOW option of the RUN command for TACL interface.

Section 2: Descriptions of new and changed resources and command line options are included. The description of the international character set supported by Tandem is added. Clarifications are made to some of the descriptions of resource and customization specifications.

Heading_1

Contents

About This Manual ix

Notation Conventions xi

Section 1 Using the x6530 Emulator

x6530 Window 1-1

Multiple x6530 Windows 1-2

Window Cursors 1-3

Sizing the x6530 Window 1-4

x6530 Menus 1-5

x6530 Main Menu 1-5

About x6530 Option 1-6

Custom Commands Menu 1-7

Configuration Info Menu 1-10

x6530 Special Features 1-14

Using the Scrollbar 1-14

Simulating Function Keys 1-16

Copying and Pasting Unprotected Text 1-17

Copying and Pasting Protected Text 1-18

Section 2 x6530 Run Command and Customizing Options

x6530 Run Command 2-1

Resource and Class Variables 2-5

Reading Tandem Commands From a File 2-8

x6530 Custom Commands Menu 2-9

Character Class for Copying and Pasting Text 2-11

How x6530 Selects a Word 2-12

Changing Character Class Definitions 2-14

Using National Languages With x6530 2-22

Keymap Files 2-23

Creating a New Keymap File 2-23

Creating a New Keymap File for a National Language 2-29

Executing the x6530 Run Command 2-31

Setting Up the UNIX Workstation	2-31
User Accounts and Path Commands	2-32
DISPLAY Variable	2-32
.Xdefaults File	2-32
Setting Up mwm's Menu	2-32
Automatically Starting mwm	2-33
Using an X Server to Start mwm	2-33
Using a Telnet X Client to Start mwm	2-33
Using xdm to Start mwm	2-33

Index	Index-1
--------------	---------

Figures	Figure 1-1.	x6530 Window	1-1
	Figure 1-2.	x6530 Main Menu	1-5
	Figure 1-3.	Custom Commands Menu	1-7
	Figure 1-4.	Emulator Options Menu	1-8
	Figure 1-5.	Configuration Info Menu	1-10
	Figure 1-6.	Resources Display	1-11
	Figure 1-7.	Keymaps Display	1-12
	Figure 1-8.	Character Classes Display	1-13
	Figure 1-9.	x6530 Scrollbar	1-14
	Figure 1-10.	PS Mail Function Key Display	1-16
	Figure 2-1.	Sample .Xdefaults file for x6530	2-5
	Figure 2-2.	Sample Keymap File (partial listing)	2-24
	Figure 2-3.	Keysym for 9 Key	2-25
	Figure 2-4.	Keysyms for Shift and 9 Keys	2-25

Tables	Table 1-1.	x6530 Menu Options	1-6
	Table 1-2.	Emulator Options Menu Choices	1-9
	Table 1-3.	Scrollbar Functions in Conversational Mode	1-15
	Table 1-4.	Scrollbar Functions in Block Mode	1-16
	Table 2-1.	x6530 Command Line Options	2-2
	Table 2-2.	Resource and Class Variables	2-6
	Table 2-3.	x6530 Default Character Classes	2-11
	Table 2-4.	Tandem Character Set	2-15
	Table 2-5.	National Languages for x6530	2-22
	Table 2-6.	6530 Keyboard Functions	2-27
	Table 2-7.	National Characters Supported by x6530	2-29
	Table 2-8.	Danish Keysyms	2-30
	Table 2-9.	x6530 ASCII Characters for Your Keys	2-30

About This Manual

This manual explains how to use and customize the x6530 Emulator, an X client that gives you access to Tandem systems from a variety of hosts within an X Window System environment, and that allows a variety of workstations to emulate a Tandem 6530 terminal.

Audience and Prerequisites

This manual is intended for end users and administrators. As an end user, you are assumed to have experience with the X Window System, the Tandem Advanced Command Language (TACL), and other Tandem software products. As an administrator, you are assumed to have experience with Guardian 90, Tandem software products, the X Window System, UNIX, and TCP/IP.

Prerequisite Equipment

This manual also assumes the user's host is a workstation, personal computer (PC), or X terminal. It assumes the host where x6530 resides is a PC or workstation running the UNIX operation system. If you have a different operating system, see your system's manuals for the proper commands and instructions to use.

Throughout this manual the term workstation is used to mean a workstation, PC, or X terminal.

Prerequisite Manuals

The x6530 Emulator follows the operating conventions of other X Window System clients. See the *X Window System User's Guide* if you are unfamiliar with the X environment or using X clients.

If you are unfamiliar with Tandem systems, you should have the following manuals:

- Introduction to Tandem NonStop Systems*
- Overview of Tandem Products for NonStop Systems*
- Guardian 90 Operating System User's Guide*

Installing x6530

Instructions for installing x6530 on a workstation are not included in this manual. All the x6530 software is provided on a tape that is standard input to the Tandem INSTALL program. After those files are copied to the Tandem host using standard installation procedures, you can print a Tandem file named INSTALL that has procedures for installing x6530 on a UNIX workstation. For example:

```
6> TGAL /IN $SYSTEM.ZX6530.INSTALL, OUT $S.#PRT1/
```

Manual Organization

This manual is organized as follows:

- Section 1 What the x6530 window looks like and how to use its features. This section also describes the standard menus of x6530.
- Section 2 How to customize the look and operation of x6530. This section also gives suggestions for executing the x6530 run command.

Notation Conventions

The following list summarizes the conventions for syntax presentation in this manual.

Notation	Meaning
UPPERCASE LETTERS	Uppercase letters represent keywords and reserved words; enter these items exactly as shown.
<i>lowercase italic letters</i>	Lowercase italic letters represent variable items that you supply.
Brackets []	Brackets enclose optional syntax items. A group of vertically aligned items enclosed in brackets represents a list of selections from which you can choose one or none.
Braces { }	Braces enclose required syntax items. A group of vertically aligned items enclosed in braces represents a list of selections from which you must choose one.
Vertical line	A vertical line separates alternative syntax items in a horizontal list. Such a list, enclosed in either brackets or braces, is an alternative to a vertical list for presenting selections.
Ellipsis ...	An ellipsis immediately following a pair of brackets or braces indicates that you can repeat the enclosed syntax items any number of times.
Spaces	If a space separates two items, that space is required. If one of the items is a punctuation symbol, such as a parenthesis or a comma, spaces are optional.
Punctuation	Parentheses, commas, semicolons, and other symbols not described above must be entered precisely as shown.
Ctrl/ <i>key-name</i>	A notation of this type means to hold down the Control key while pressing the specified key.

1 Using the x6530 Emulator

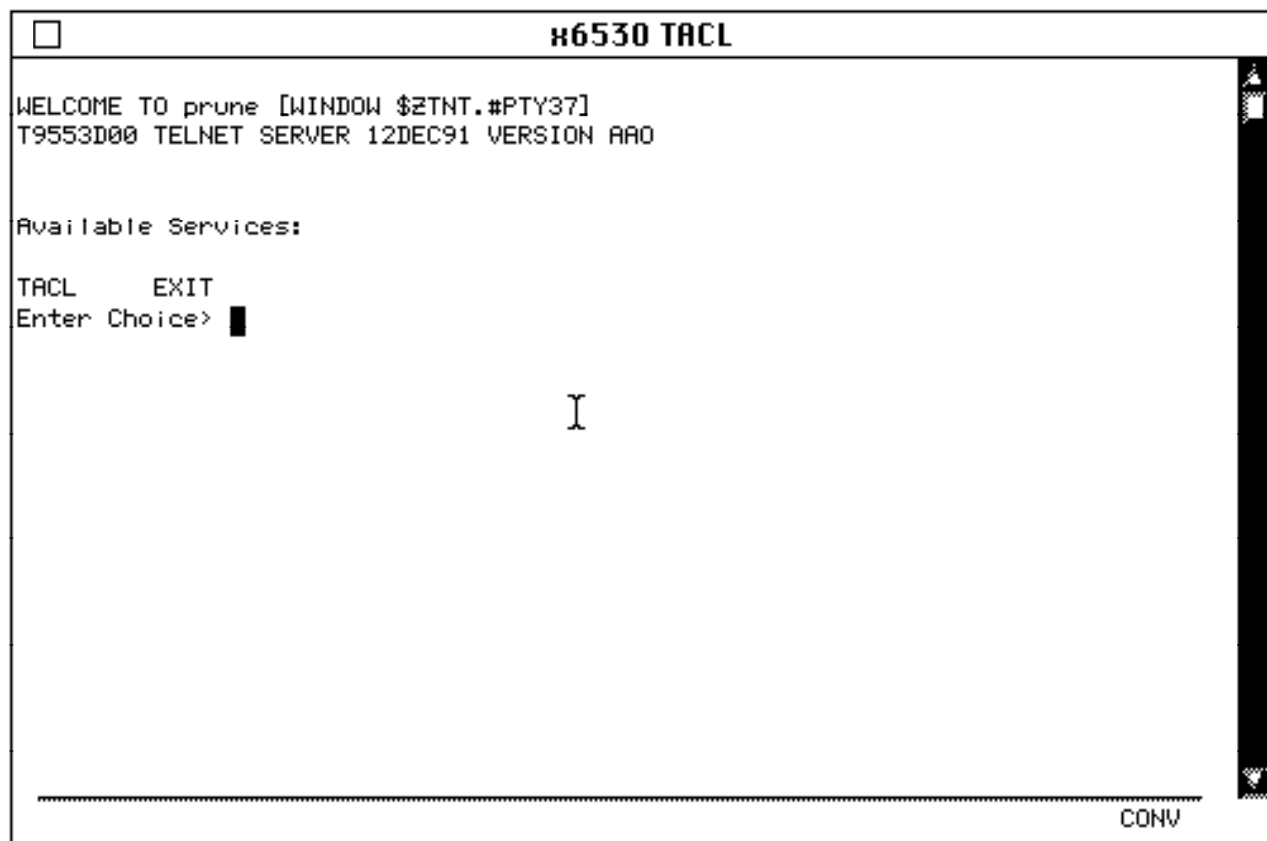
The x6530 Emulator is a terminal emulator in the X Window System environment that gives you access to a Tandem system from your workstation. It is analogous to xterm, the terminal emulator described in the *X Window System User's Guide*.

When you have an x6530 window displayed at your workstation, you can log on to a Tandem host to execute any program or TACL command residing there. You can also have more than one x6530 window displayed at your workstation at a time.

This section assumes that you or your system administrator has set up x6530 to run with the command line options and other customizing options you want. It also assumes that the x6530 run command was executed by one of the methods described in Section 2, "x6530 Run Command and Customizing Options."

x6530 Window When x6530 is running, an x6530 window is displayed at your workstation. Figure 1-1 shows the window as it appears with no customization.

Figure 1-1. x6530 Window



001

Note The window shown in Figure 1-1 is a generalization. The actual appearance of the window depends on the specific window-manager software in use, so details may vary.

To access the Tandem host, follow these steps:

1. Enter `tacl` at the `Enter Choice>` prompt to get to the TACL prompt.
2. Log on to the Tandem host when the TACL prompt appears. Logging on requires a Tandem logon ID and password.
3. Enter a TACL command or the name of a Tandem program. (Tandem programs and TACL commands can also be selected from an x6530 custom menu. See Section 2, "x6530 Run Command and Customizing Options," for information about creating and using an x6530 custom menu.)

When you have logged on to the Tandem host, you have started an x6530 session. When you no longer want to use the x6530 window, you can do one of three things:

- Stop x6530 with the Quit option of the x6530 menu. (See "x6530 Menus," later in this section, for information on the x6530 menu.)
- Stop x6530 by logging off from the Tandem host.
- Turn the window into an icon from your window manager.

Stopping x6530 ends the session by logging you off the Tandem host and erasing the window from your screen. Turning the window into an icon does not log you off from the Tandem host, preserving the session for later use.

Multiple x6530 Windows There are two ways to have more than one x6530 window at your workstation at the same time:

- Use the WINDOW run option when executing a Tandem program during an x6530 session.
- Run x6530 as many times as you need windows.

WINDOW Run Option

After you have started an x6530 session, you can run a Tandem program in a separate x6530 window at your workstation by using the WINDOW run option in the command that starts the Tandem program. WINDOW can appear in any order with the program's other run options. The program is the only process in the new window; a new TACL process is not started.

For example, if you wanted Peruse to run in a separate window at your workstation, you would enter a command similar to the following at the TACL prompt in the first x6530 window:

```
8> PERUSE /WINDOW "-title Peruse"/
```

When you exit from Peruse, the x6530 window it was running in disappears and the x6530 process associated with Peruse is stopped. The original x6530 window remains at your workstation until you stop it from the x6530 menu or log off the Tandem host.

Having Tandem programs run in separate x6530 windows is a convenient way to use them. You can switch back and forth between the windows rather than interrupt your work in one x6530 window to start another program. For example, you could have Peruse, FUP, and PS Mail running in x6530 windows of their own, keeping the original x6530 window for other tasks. When you needed to check your mail, you would place the pointer in the window containing PS Mail, use PS Mail, then move the pointer back to the original x6530 window to continue working there.

Using the WINDOW run option to create multiple x6530 windows is preferable to running x6530 again because:

- The easiest way to start a new x6530 window is with the WINDOW run option.
- There is less overhead when TACL is not running in the new window.
- The same DEFINES, ASSIGNS, and so on, from the original TACL process are in effect for the new window.
- The x6530 command line options used for the original x6530 window are used for the new x6530 window, with the exceptions of -exec, -dexec, -title, -n, and -i.
- The new window automatically disappears and the x6530 process for it stops when you stop the program.

Running x6530 Multiple Times

You can have multiple x6530 windows by executing an x6530 run command as many times as you would like windows at your workstation. In this case, you will have a TACL process in each window. This method is the same as logging on to a Tandem host at more than one 6530 terminal. When you end one of several x6530 sessions, you are automatically logged off the Tandem host for that window's session. The other x6530 sessions remain active.

This method of multiple windows greatly increases the computer overhead because a TACL process is started for each window. In addition, it is not as easy to get rid of the extra windows as it is when windows are created with the WINDOW run option. You must log off the Tandem host or close each window from the x6530 menu when you are finished with it.

Window Cursors There are two types of window cursors:

- Pointer cursor (also called the mouse cursor); it looks like the letter "I" within an x6530 window.
- Text cursor (also called the 6530 cursor); it looks like a solid rectangle in an active window, a hollow rectangle in an inactive window.

Both cursors are shown in Figure 1-1.

Pointer Cursor

The pointer cursor is used to:

- Track the movement of your mouse around the screen.
- Manipulate windows and icons.
- Select items in menus.
- Select the window in which you want to enter characters.

Note This manual assumes you have a three-button mouse attached to your workstation. The buttons are referred to as the right mouse button, middle mouse button, and left mouse button. If you have a different type of mouse device, see your workstation's documentation about simulating a three-button mouse.

Text Cursor

The text cursor defines the location in a window where you want to enter characters or move characters. To position the text cursor:

1. Move the pointer to the desired location in the window.
2. Click the left mouse button to move the text cursor to the position of the pointer.

This operation has no effect, however, if the pointer is on line 25 of the window.

In a Tandem block-mode application, you can quickly move the text cursor forward and backward through a line of text by using these keystrokes:

- | | |
|------------------|---|
| Ctrl/Right Arrow | Moves the text cursor to the next word to the right of the cursor's present position. |
| Ctrl/Left Arrow | Moves the text cursor to the next word to the left of the cursor's present position. |

Note Tandem block-mode applications often have protected fields where the text cursor is not allowed. If you click the left mouse button while the pointer is on a protected field, the text cursor moves to the preceding unprotected field.

Sizing the x6530 Window Initially, x6530 starts in a window with enough room for 24 lines and 80 columns (characters) of text. The window can be made smaller or larger for a Tandem application operating in conversational mode.

Do not resize the x6530 window when you are running a Tandem block-mode application.

x6530 Menu The x6530 Emulator has one standard pop-up menu, with three additional submenus that cascade down from it: the Custom Commands, Emulator Options, and Configuration Info menus. These menus are described in this subsection.

When using the x6530 menus, the pointer changes to an arrow (different from the alternative pointer that can be configured for x6530 operations). On returning from the menus to the x6530 window, the pointer reverts to its default shape.

x6530 Main Menu The x6530 main menu is shown in Figure 1-2.

Figure 1-2. x6530 Main Menu



002

The x6530 main menu provides options to get information about the current x6530 session, stop an x6530 session, reset the x6530 window without stopping the session, or select any of the three submenus. To pop up this menu and select an option:

1. Move the pointer to any place inside the x6530 window.
2. Hold down the right mouse button while dragging the pointer to the option you want. If the option brings a submenu into view, move the pointer into the submenu, then continue down to the option you want in that menu. When an option is selected, it takes on a three-dimensional appearance.
3. Release the mouse button to execute the option.

Another way to pop up this menu and select an option is as follows:

1. Place the pointer anywhere in the x6530 window and click the right mouse button.
2. Move the pointer to the desired option or menu title and click the right mouse button again.
3. If a submenu appears, move the pointer to the desired option in that menu and click the right mouse button a third time.

The x6530 menu options are described in Table 1-1.

Table 1-1. x6530 Menu Options

Option	Description
About x6530	Brings up a text window showing version information about x6530 and about the X server that is currently running.
Custom Commands	Cascades to a menu containing the names of Tandem programs that can be started simply by selecting them from this menu.
Emulator Options	Cascades to a menu containing options that affect the appearance and actions of the x6530 window.
Configuration Info	Cascades to a menu containing options for displaying the resources, keymaps, and character classes in effect for x6530.
Reset	Clears and internally resets the x6530 window. Press the Return or Enter key to return to the process that was running when a reset was done.
Quit	Stops x6530, erases the x6530 window, and stops the Tandem process that was running in the window. If you are running a program other than TACL, you should exit from the program normally instead of using the Quit option; otherwise, you may not exit cleanly.

About x6530 Option

The About x6530 option brings up a text window that describes the version of x6530, and the X server, that is currently running. The display is similar to the following:

```

x6530 version      : C32c 24FEB92
System name       : guido
OS version        : UMIPS
OS release        : 4_52
Machine type      : mips
Display name      : decay:0.0
XServer vendor    : MacX 1.1.7: Apple Computer, Inc.
XServer version   : 11
XServer release   : 4
  
```

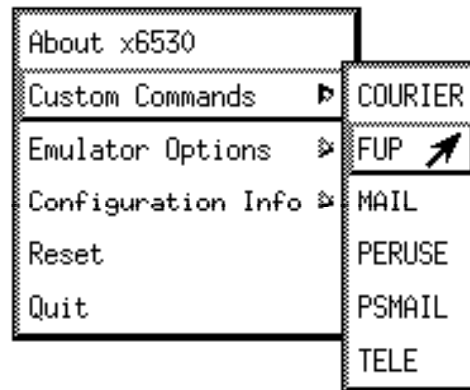
OK

003

To close the display and return to the x6530 window, position the pointer on the “OK” button and click the left mouse button. Alternatively, you can press the Return or Enter key.

Custom Commands Menu The Custom Commands menu, shown in Figure 1-3, provides a quick way of starting Tandem programs. To run a program, simply select its name from this menu.

Figure 1-3. Custom Commands Menu

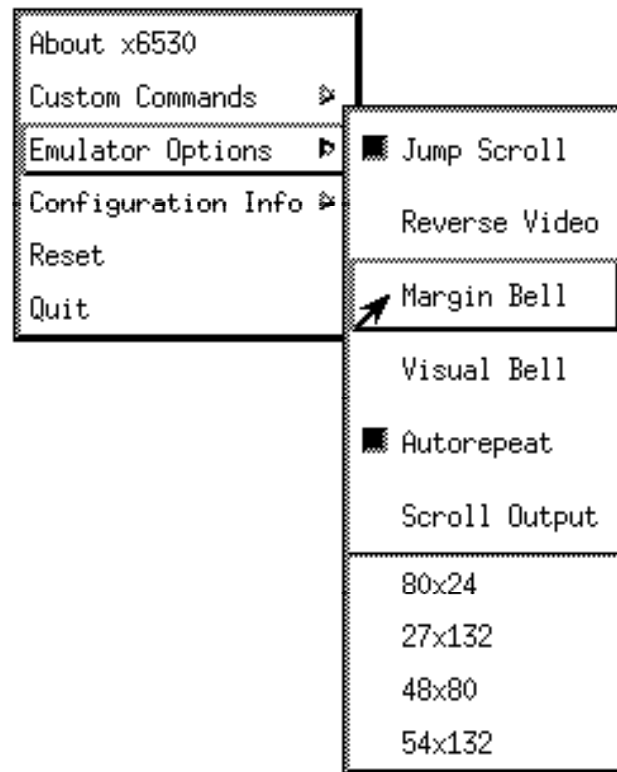


004

The contents of this menu are defined by the `.x6530rc` file. See Section 2, “x6530 Run Command and Customizing Options,” for a description of the entries in that file.

Emulator Options Menu The Emulator Options menu allows you to change options that affect the appearance of the x6530 window and the way x6530 handles text. You can switch these options on and off during the current session; the changes you make are in effect for the current session only. The Emulator Options menu is shown in Figure 1-4.

Figure 1-4. Emulator Options Menu



005

The Jump Scroll, Reverse Video, Margin Bell, Visual Bell, Autorepeat, and Scroll Output options are toggles; selecting one of these options switches its state from off to on, or from on to off. A “pushbutton” (a square with a three-dimensional appearance) is placed next to an option when it is on; for example, in Figure 1-4, two options are on and a third will be turned on as soon as the mouse button is released.

The window size options are mutually exclusive; only one window size can be in effect at a given time. Turning on a window size specification turns off the one that had been in effect previously. In conversational mode, you have four screen sizes from which to choose; in block mode, you have only one choice, determined by the screenType resource.

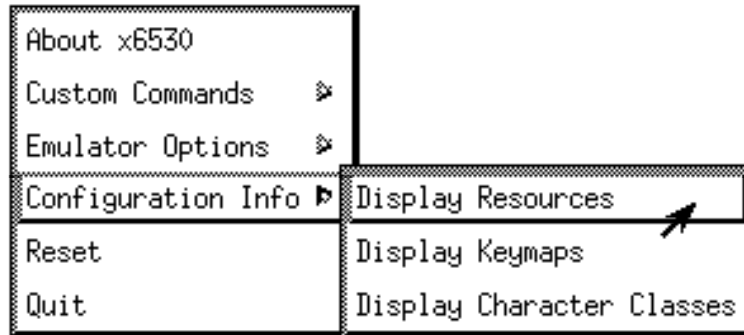
The Emulator Options menu choices are described in Table 1-2.

Table 1-2. Emulator Options Menu Choices

Option	Description																
Jump Scroll	Turns on jump scroll, a faster method of scrolling lines in a window. Using this menu option is the same as using the command line option <code>-j</code> (see Section 2).																
Reverse Video	Reverses the foreground and background colors. Using this option is the same as using the command line option <code>-rv</code> (see the <i>X Window System User's Guide</i>).																
Margin Bell	Rings a margin bell when you type near the right margin. Using this menu option is the same as using the command line option <code>-mb</code> (see Section 2).																
Visual Bell	Causes the window to flash when you type near the right margin instead of ringing a bell. The Margin Bell option must be turned on for Visual Bell to work. Using this menu option is the same as using the command line options <code>-vb</code> and <code>-mb</code> together (see Section 2).																
Autorepeat	<p>Causes keys that otherwise would not have an autorepeat function to send repeated key signals as long as the key is held down. On an actual 6530 terminal, the function keys (F1–F16 and SF1–SF16), and the following keys, do not repeat:</p> <table border="0"> <tbody> <tr> <td>CLEARALLTABS</td> <td>ENDOFFPAGE</td> <td>INSERTMODE</td> <td>SETTABS</td> </tr> <tr> <td>CLEARTABS</td> <td>FIRSTPAGE</td> <td>LASTPAGE</td> <td>SHIFTEDROLLDN</td> </tr> <tr> <td>DELETELINE</td> <td>HOME</td> <td>PGDN</td> <td>SHIFTEDROLLUP</td> </tr> <tr> <td>END</td> <td>INSERTLINE</td> <td>PGUP</td> <td>STARTOFLINE</td> </tr> </tbody> </table> <p>When Autorepeat is on (the default), all of these keys repeat while held down.</p>	CLEARALLTABS	ENDOFFPAGE	INSERTMODE	SETTABS	CLEARTABS	FIRSTPAGE	LASTPAGE	SHIFTEDROLLDN	DELETELINE	HOME	PGDN	SHIFTEDROLLUP	END	INSERTLINE	PGUP	STARTOFLINE
CLEARALLTABS	ENDOFFPAGE	INSERTMODE	SETTABS														
CLEARTABS	FIRSTPAGE	LASTPAGE	SHIFTEDROLLDN														
DELETELINE	HOME	PGDN	SHIFTEDROLLUP														
END	INSERTLINE	PGUP	STARTOFLINE														
Scroll Output	Causes x6530 to scroll to the last page whenever a key is pressed or new output is received from the Tandem host. This is useful if you use the Page keys to look back in the save buffer but want to make sure that new output is appended to the bottom. Caution: this option can produce undesirable results when used with conversational-mode programs that use the full screen instead of writing only to the end of it.																
Window size options	Selecting the 80 x 24 option returns the x6530 window to its default size of 80 characters wide by 24 lines high. The other size options—27 x 132, 48 x 80, and 54 x 132—change the window size to the stated dimensions.																

Configuration Info Menu The Configuration Info menu lets you display any of three sets of information about the x6530 configuration. The Configuration Info menu is shown in Figure 1-5.

Figure 1-5. Configuration Info Menu

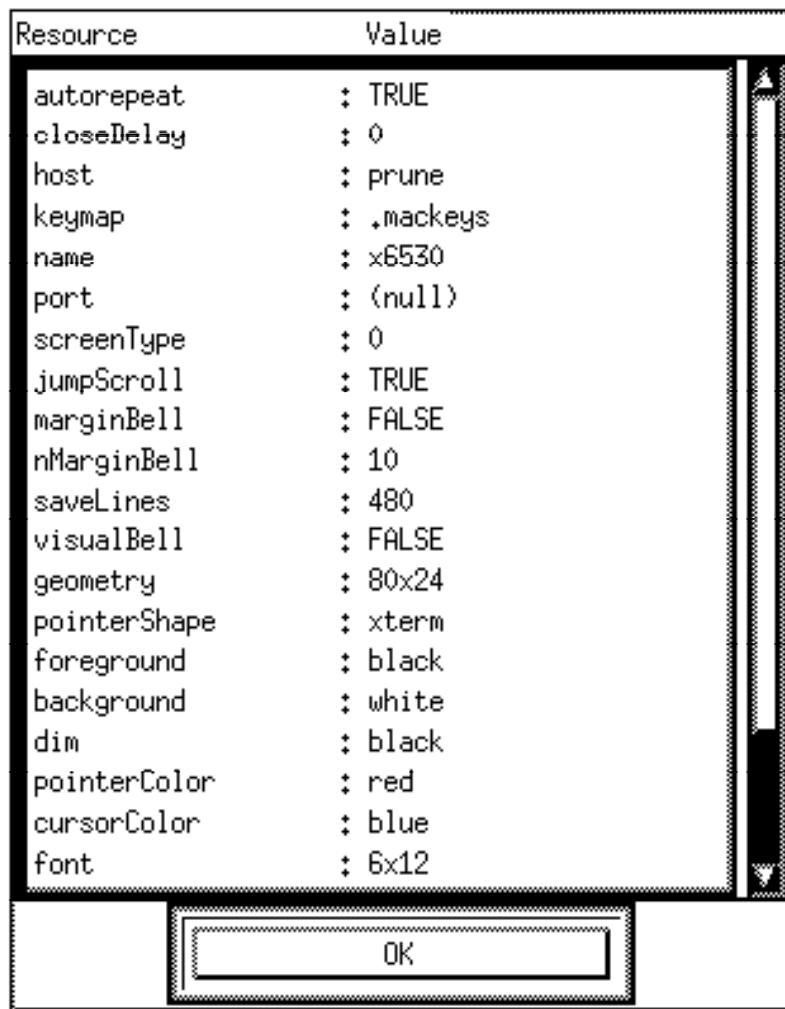


006

Display Resources Option

The Display Resources option pops up a text window that shows the status of every option that can be configured, such as the display font, the name of the keymap file in use, the name of the custom command file, the shape of the pointer, and so on. Figure 1-6 is an example of this display.

Figure 1-6. Resources Display



The screenshot shows a window titled "Resources Display" with a list of resources and their values. The window has a title bar and a scroll bar on the right. At the bottom, there is an "OK" button.

Resource	Value
autorepeat	: TRUE
closeDelay	: 0
host	: prune
keymap	: .mackeys
name	: x6530
port	: (null)
screenType	: 0
jumpScroll	: TRUE
marginBell	: FALSE
nMarginBell	: 10
saveLines	: 480
visualBell	: FALSE
geometry	: 80x24
pointerShape	: xterm
foreground	: black
background	: white
dim	: black
pointerColor	: red
cursorColor	: blue
font	: 6x12

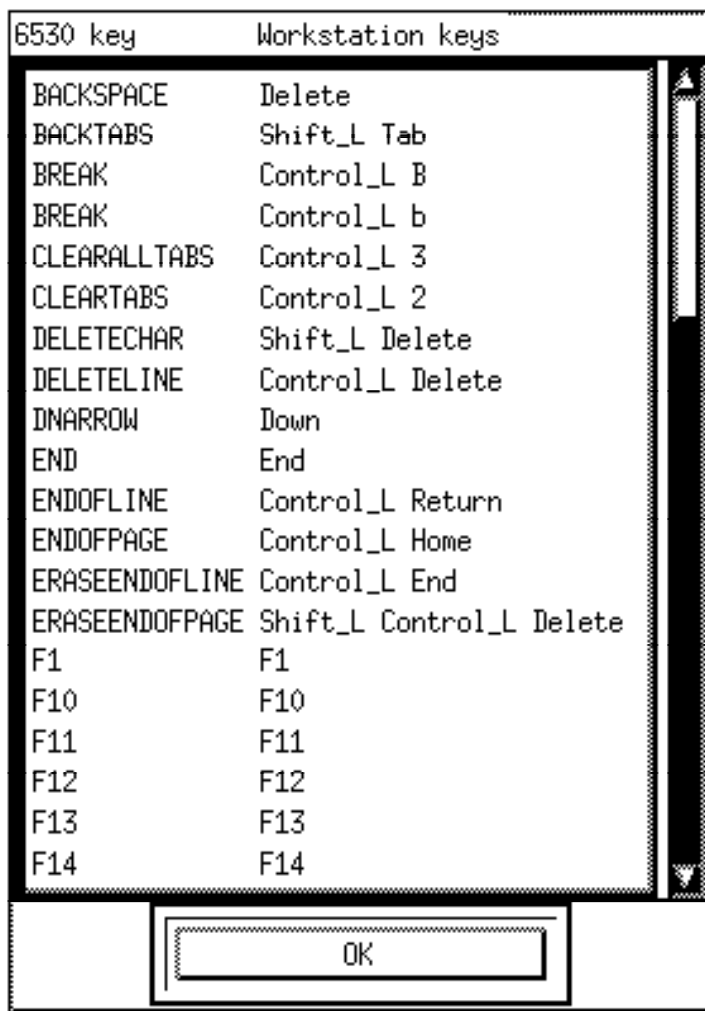
007

To close the display and return to the x6530 window, click on the "OK" button. Alternatively, you can press the Return or Enter key.

Display Keymaps Option

The Display Keymaps option pops up a text window that shows the current mapping of every 6530 key. The 6530 keys are listed in alphabetic order for quick reference. Some 6530 keys may be mapped to more than one workstation key; an additional line is displayed for each additional definition. Figure 1-7 is an example of a keymaps display.

Figure 1-7. Keymaps Display



6530 key	Workstation keys
BACKSPACE	Delete
BACKTABS	Shift_L Tab
BREAK	Control_L B
BREAK	Control_L b
CLEARALLTABS	Control_L 3
CLEARTABS	Control_L 2
DELETECHAR	Shift_L Delete
DELETELINE	Control_L Delete
DNARROW	Down
END	End
ENDOFLINE	Control_L Return
ENDOFFPAGE	Control_L Home
ERASEENDOFLINE	Control_L End
ERASEENDOFFPAGE	Shift_L Control_L Delete
F1	F1
F10	F10
F11	F11
F12	F12
F13	F13
F14	F14

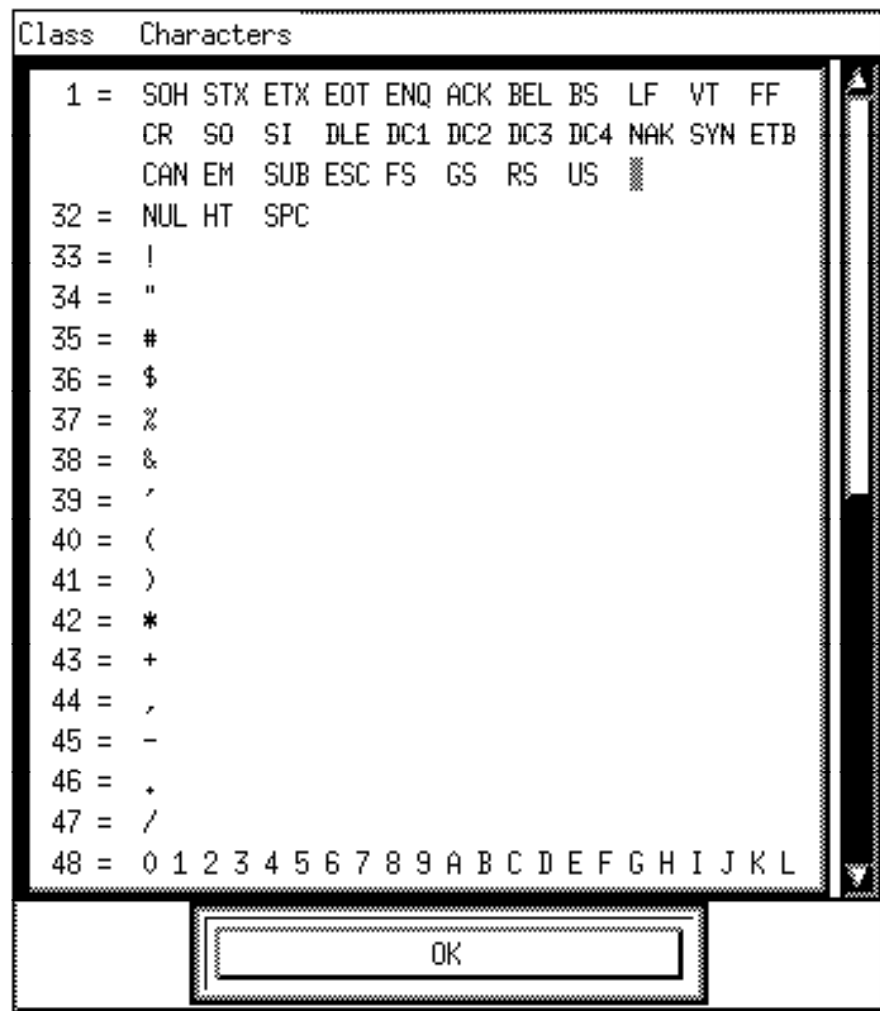
008

To close the display and return to the x6530 window, click on the “OK” button. Alternatively, you can press the Return or Enter key.

Display Character Classes Option

The Display Character Classes option pops up a text window that shows the members of each character class for reference purposes. Figure 1-8 is an example of a character classes display.

Figure 1-8 . Character Classes Display



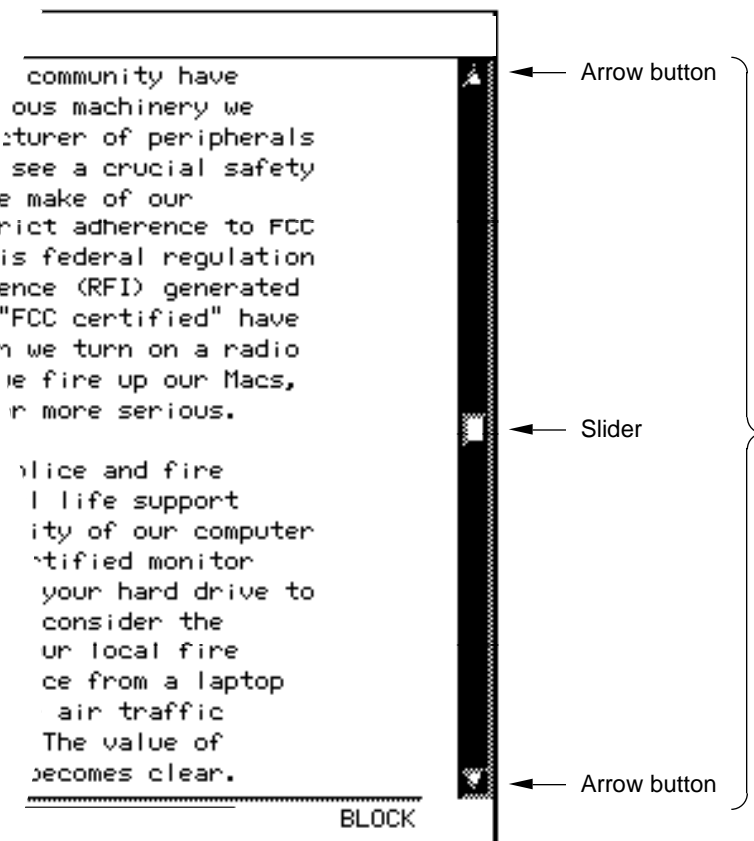
009

To close the display and return to the x6530 window, click on the "OK" button. Alternatively, you can press the Return or Enter key.

- x6530 Special Features** The x6530 Emulator has three special features associated with a mouse or other pointing device. These features include:
- Scrolling and paging through text with the scrollbar.
 - Simulating a 6530 function key.
 - Copying and pasting text within the same x6530 window or between windows.

Using the Scrollbar The scrollbar consists of a narrow rectangle called the scroll region, with upward-pointing and downward-pointing arrows (which have pushbutton functions when you click on them with the mouse button) at opposite ends of the rectangle. A smaller, contrasting rectangle called a slider is also contained within the scroll region. The scrollbar is shown in Figure 1-9:

Figure 1-9. x6530 Scrollbar



010

Using the Scrollbar in Conversational Mode

In conversational mode, the position of the slider within the scroll region represents the position within the work area of the text that is currently visible in the window. The size of the slider in relation to the scroll region represents the amount of visible text in relation to the total amount of text in the work area. For example, if the slider occupies 10 percent of the scroll region, 10 percent of the text in the work area is visible in the window; if the slider is half as large as the scroll region, the window is displaying half of all the text in the work area.

The arrow buttons, the slider, and the areas of the scroll region above and below the slider can be used to scroll and page the window backward and forward through the work area. These functions are described in Table 1-3:

Table 1-3. Scrollbar Functions in Conversational Mode

Function	Mouse Action
Scroll up or down	Place the pointer on the up or down arrow button and click the left mouse button; the window moves up or down one increment. Holding down the mouse button causes scrolling to continue until the button is released.
Page up or down	Place the pointer in the scroll region above or below the slider and click the left mouse button; the window moves up or down one page (determined by the window size). Holding down the mouse button causes paging to continue until the button is released.
Slide text up or down	Place the pointer on the slider, hold down the middle mouse button, and move the slider up or down; text scrolls rapidly in the appropriate direction.
Jump to location	Place the pointer in the area of the scroll region that represents the location of the desired text and click the middle mouse button; the slider jumps to the location of the pointer and the window displays the text at that location in the work area.
Jump to top of work area	Place the pointer on either the up arrow button or the upper area of the scroll region, hold down the Control key, and click the left mouse button.
Jump to bottom of work area	Place the pointer on either the down arrow button or the lower area of the scroll region, hold down the Control key, and click the left mouse button.

Using the Scrollbar in Block Mode

In block mode, x6530 is unable to determine how many pages are available to the block-mode program, so it cannot expand, contract, or position the slider to reflect the size and location of the visible text in relation to the total text in the work area. In block mode, the slider is always in the center of the scroll region and is reduced to its smallest size. You can use the slider for rapid scrolling as in conversational mode, but it returns to the center of the scroll region when you release the mouse button.

Many Tandem block-mode applications use paging keys on the 6530 terminal to view text that has scrolled off the screen. Table 1-4 shows the use of the scrollbar to simulate the actions of the paging keys:

Table 1-4. Scrollbar Functions in Block Mode

6530 Key	Mouse Action
RollUp	Place the pointer on the down arrow button and click the left mouse button. Hold down the mouse button to continue scrolling.
RollDown	Place the pointer on the up arrow button and click the left mouse button. Hold down the mouse button to continue scrolling.
PageUp	Place the pointer in the scroll region above the slider and click the left mouse button. Hold down the mouse button to continue paging.
PageDown	Place the pointer in the scroll region below the slider and click the left mouse button. Hold down the mouse button to continue paging.
FirstPage	Place the pointer on either the up arrow button or the upper area of the scroll region, hold down the Control key, and click the left mouse button.
LastPage	Place the pointer on either the down arrow button or the lower area of the scroll region, hold down the Control key, and click the left mouse button.

Simulating Function Keys In many Tandem block-mode applications, such as PS Mail, a description of the active function keys appears somewhere on the screen; for example, they appear across the bottom of the PS Mail main menu screen as shown in Figure 1-10.

Figure 1-10. PS Mail Function Key Display

```

F1=Scan  F2=Read  F6=Delete  F7=Add  F8=File  SF8=Format
F9=Print SF9=Copy F10=Create F13=DLists SF13=Directory F14=Profile
F15=Help SF15=Extras F16=Folders SF16=Exit

```

011

To simulate a function key, place the pointer cursor on any character in the description of that function key and click the left mouse button. For example, as shown in Figure 1-10, placing the pointer on the letter *c* in the `F1=Scan` function description, then clicking the left mouse button, sends an F1 key code to the Tandem host.

Some Tandem applications display active function key descriptions with a space on either side of the equal sign, such as:

```
F1 = Logon
```

For these types of display, you must place the pointer over the function key name (F1, F2, and so on). Nothing happens when the mouse button is clicked with the pointer over the = character or the word `Logon` because of the embedded spaces.

**Copying and Pasting
Unprotected Text**

Text from one x6530 window can be copied to another window, or copied within the same window. The first step is selecting the text you want pasted elsewhere (selecting text also copies it into the select buffer, ready for pasting in a new location). There are four ways in which unprotected text can be selected:

- By word
- By line
- By page
- By block

Selecting Text By Word

To select the text by word:

1. Position the pointer cursor on any character in the word you want to copy.
2. Double-click the left mouse button. The word that was highlighted is saved, ready for pasting elsewhere.

Note The characters selected in the word depend on a character class definition. See Section 2, "x6530 Run Command and Customizing Options," for details on character classes.

Selecting Text By Line

To select the text by line:

1. Position the pointer cursor on any character in the line you want to copy.
2. Triple-click the left mouse button. The line of text that was highlighted is saved, ready for pasting elsewhere.

Selecting Text By Page

To select the text by page:

1. Position the pointer cursor on any character in the page you want to copy.
2. Quadruple-click the left mouse button. The page of text that was highlighted is saved, ready for pasting elsewhere.

Selecting Text By Block

There are two ways to select blocks of text:

- By dragging the pointer over the text you want.
- By defining a beginning and ending point around the text you want.

To select the text by dragging the pointer cursor:

1. Position the pointer over the first character to copy.
2. Hold down the left mouse button; do not release it until you have selected all the characters to be copied.
3. Drag the pointer over the characters to copy. As the pointer is dragged, the selected characters are highlighted.
4. When you have selected all the text to copy, release the left mouse button. The text that was highlighted is saved, ready for pasting elsewhere.

To select the text by defining a beginning and ending point:

1. Move the pointer cursor to the beginning of the text to copy.
2. Click the left mouse button; this establishes the “mark,” or anchor point. It also moves the text cursor to this position.
3. Move the pointer to the end of the text to copy.
4. Hold down the Shift key and click the left mouse button to highlight all the characters from the mark to the current pointer location.

You can also use the mark and shift-click method to deselect text that was previously selected by the same method.

Note The x6530 Emulator cannot select text that spans multiple pages of a block-mode application, such as TEDIT (though TEDIT has its own internal mechanism for selecting large areas of text); it can select only the text on the currently displayed page. The x6530 Emulator can, however, select text from more than one page in the conversational mode, where scrolling is possible.

Pasting Selected Text

To paste selected text:

1. Move the text cursor to the window position where you want to paste the previously copied text.
2. Click the middle mouse button to paste the text.

Copying and Pasting Protected Text

The preceding copying and pasting procedures copy any characters that are not in protected areas of the screen. If you need to copy characters in protected areas, follow the preceding selection steps, but select the text by holding down the Ctrl (control) key in addition to the other keys specified.

When positioning the mark in a protected field, clicking the left mouse button moves the text cursor to the unprotected field preceding the pointer position; the text cursor cannot be positioned in a protected field.

Paste the selected text as described previously.

2 x6530 Run Command and Customizing Options

This section is for system administrators, experienced with the X Window System and UNIX, who will set up x6530 for users. References to “user” mean a person using a workstation with an X server who may be running x6530 locally or who may access a remote machine to run x6530; references to “you” mean the system administrator.

Running x6530 means executing a run command at the UNIX workstation where x6530 resides. This section describes the format and options of the x6530 run command, and how to conveniently issue the run command from workstations on the TCP/IP network.

Like other X clients, x6530 can be customized. You can use any combination of the following methods to change the way x6530 runs, the way it displays the window, and the way it handles the characters inside the window:

- Use command line options with an x6530 run command.
- Use resource and class variables.
- Use an x6530 custom menu.

Most command line options, resource and class variables, and the x6530 custom menu are explained in the following subsections.

x6530 Run Command You run x6530 by issuing a command at the workstation where it resides. The format of the run command is:

```
x6530 host-name [ option ... ]
```

host-name

is the name of the Tandem host the user wants to access. This name is defined in the TCP/IP HOSTS file. See your network administrator for valid names.

option

is any of the standard X Toolkit command line options and those listed in Table 2-1. Most options can also be stored as resource or class variables in the .Xdefaults file in a user’s home directory on the workstation where x6530 resides; these options are automatically set when x6530 is started (see “Resource and Class Variables,” later in this section). For more information on the X Toolkit standard command line options, see the *X Window System User’s Guide*.

Table 2-1. x6530 Command Line Options (Page 1 of 2)

Option	Description
-auto	Turns on the autorepeat function for keys that do not, on an actual 6530 terminal, repeat their key signal while held down. This option can also be turned on from the Emulator Options menu (see Section 1). The default is on.
+auto	Turns off the autorepeat function.
-cc <i>range:class</i>	Defines a character class used in the selection process of a copy and paste operation when text is chosen by words. Which characters are selected depends on a character class definition. See "Character Class for Copying and Pasting Text," later in this section, for details.
-closeDelay <i>n</i>	Specifies the number of seconds to wait after the end of a session for x6530 to close the window (to allow time for reading any messages that might appear if an error terminates the session). The default is zero. A value of -1 causes x6530 to wait indefinitely.
-cr <i>color</i>	Specifies the color for the text cursor (shaped like a rectangular box). The standard X colors are listed in the <i>X Window System User's Guide</i> . The default is black.
-dexec <i>file</i>	Directs x6530 to read commands initially from <i>file</i> instead of from the keyboard. After <i>file</i> is read it is deleted. (See "Reading Tandem Commands From a File," later in this section, for details.)
-dim <i>color</i>	Sets the color for the 6530 dim video attribute. The standard X colors are listed in the <i>X Window System User's Guide</i> . The default is red.
-exec <i>file</i>	Tells x6530 that commands should initially be read from <i>file</i> instead of the keyboard. This command is the same as -dexec except that <i>file</i> is not deleted after it is read. (See "Reading Tandem Commands From a File," later in this section, for details.)
-fb <i>font</i>	Specifies the font to use when displaying bold text. If no bold font is specified, the normal font is used (drawn twice and separated by one pixel). The x6530 Emulator uses bold text to represent blinking text. The standard X fonts are listed in the <i>X Window System User's Guide</i> .
-fn <i>font</i>	Specifies the font to use for normal text. The standard X fonts are listed in the <i>X Window System User's Guide</i> . The default is 9x15.
-help	Displays a summary of x6530 command line options in the xterm window before displaying the x6530 window.
-host <i>host</i>	Specifies the name of the Tandem host to which x6530 is to be connected. There is no default for this option.
-iconic	Starts x6530 as an icon. The default is off.
-j	Turns on jump scroll, a faster method of scrolling lines in a window. This option can be turned on and off from the Emulator Options menu (see Section 1). The default is on.
+j	Turns off jump scroll.

Table 2-1. x6530 Command Line Options (Page 2 of 2)

Option	Description
-keymap <i>file</i>	Specifies the keymap file to use; for example, if x6530 is running on workstation A and displaying on workstation B, the keymap should correspond to the keyboard and server on workstation B. The x6530 Emulator searches for the keymap file on workstation A. If no keymap option is used, x6530 uses the file <code>.x6530modkey</code> in the user's home directory or in the <code>/etc</code> directory on the workstation it is running on. See "Keymap Files," later in this section, for details.
-mb	Rings a margin bell when you type near the right end of a line. This option can be turned on and off from the Emulator Options menu (see Section 1). The default is on.
+mb	Turns off the margin bell.
-ms <i>color</i>	Specifies the color for the pointer (shaped like the letter "I"). The standard X colors are listed in the <i>X Window System User's Guide</i> . The default is black.
-n <i>icon-name</i>	Specifies the icon name when the x6530 window is turned into an icon. The default name is x6530.
-nb <i>number</i>	Specifies the number of characters from the right end of a line at which the margin bell rings or the screen flashes. The default is 10.
-port <i>n</i>	Specifies the port number of the Telserv connection. By default, x6530 uses the default port set up for your system.
-screenType <i>n</i>	Specifies the size of the screen used when x6530 is in block mode. You can specify one of four values for <i>n</i> : 0 specifies a screen size of 24 lines by 80 columns; 1 specifies 27x132, 4 specifies 48x80, and 5 specifies 54x132. The default is 0.
-sl <i>number</i>	Specifies the number of lines to save that scroll off the top of a window. You can specify between 240 and 2160 lines; the default is 480. Any number below 240 or above 2160 is treated as 240 or 2160, respectively.
-T <i>window-title</i>	Specifies a window title used by a window manager. This title is used only when a window manager places a title banner around a window. <i>Window-title</i> must be surrounded by double quotation marks if it is more than one word. The default name is x6530.
-vb	Specifies that instead of ringing the terminal bell when you type close to the right end of a line, the window flashes. You must use -mb with -vb to have -vb work. This option can be turned on and off from the Emulator Options menu (see Section 1). The default for -vb is off.
-xrm <i>resource-spec</i>	Permits the setting, on the command line, of any resource that can be set in the <code>.Xdefaults</code> file. The format for <i>resource-spec</i> is <code>'resource-name:value'</code> (single quotes are required).

If an error conditions arises, x6530 displays an error message in the xterm window at the workstation where x6530 resides, not in your x6530 window.

See "Executing the x6530 Run Command," later in this section, for suggestions about issuing an x6530 run command.

Examples

1. The following run command connects x6530 to the Tandem host named “madcap.” The x6530 window displays its output on a workstation named “frank,” defined by the -display option.

```
x6530 madcap -display frank:0 &
```

2. The following run command connects x6530 to the Tandem host named “zip.” The x6530 window displays its output on a workstation named “jack,” defined by the -display option. A -keymap option was used to define a keymap file because the default keymap file .x6530modkey does not work for the X server and keyboard of jack.

```
x6530 zip -display jack:0 -keymap /etc/.xkmPCXView &
```

Resource and Class Variables

The x6530 Emulator uses the resource and class variables described in Table 2-2. These items can be stored in a file named `.Xdefaults` in the user's home directory on the workstation where x6530 resides so they are automatically set when x6530 is run. Use the variables for x6530 by prefixing them with x6530 as shown in Figure 2-1.

Depending on your workstation, other files, in addition to or instead of the `.Xdefaults` file, may be consulted to define the resources for an X application. For example, the UNIX variable `XENVIRONMENT` can be used to specify a different file. Consult your workstation documentation and the *X Window System User's Guide* for additional information. Although different files may be used, resource variables are specified in the format shown here for the `.Xdefaults` file.

Resource and class variables overlap with x6530 command line options and Emulator Options menu choices. Depending on users' preferences, you could use all three methods to customize x6530. Perhaps all users' `.Xdefaults` files would have the same set of resource names or classes; their run commands would vary depending on command line options, and, finally, those few options left over would be left to each user to control during a session with the x6530 Modes menu.

You should provide a default `.Xdefaults` file for your users. Users can customize their `.Xdefaults` files or use command line options to personalize their environment. The Emulator Options menu is generally used to experiment with personalization within a session; desirable options would then normally be made permanent by adding them to the `.Xdefaults` file.

For more details on defining resource and class variables, see the *X Window System User's Guide*. A sample `.Xdefaults` file is shown in Figure 2-1.

Figure 2-1. Sample `.Xdefaults` file for x6530

```
x6530.*.background:    blue
x6530.*.foreground:    white
x6530.*.borderColor:  red
x6530.*.borderWidth:  2
x6530.*.font:         fixed
x6530.*.dim:         yellow
x6530.*.boldFont:    bogus
x6530.*.jumpScroll:  on
x6530.*.scrollBar:   on
x6530.*.banner:      off
x6530.*.saveLines:   512
x6530.*.visualBell:  off
x6530.*.marginBell:  on
x6530.*.cursorColor: white
x6530.*.pointerColor: white
```

012

Table 2-2. Resource and Class Variables (Page 1 of 2)

Resource (Class)	Description
autorepeat (Autorepeat)	Turns on the autorepeat function for keys that do not, on a 6530 terminal, repeat their key signal while held down. This option can also be turned on from the Emulator Options menu (see Section 1). The default is on.
background (Background)	Specifies the background color of the window. The default is gray.
boldFont (Font)	Specifies the name of the bold font.
borderWidth (BorderWidth)	Specifies the width of the window border in number of pixels. The default is 2.
borderColor (Foreground)	Specifies the border color. The default is black.
charClass (CharClass)	Defines a character class used in the selection process of a copy and paste operation when text is chosen by words. Which characters are selected depends on a character class definition. See "Character Class for Copying and Pasting Text," later in this section, for details.
closeDelay (CloseDelay)	Specifies the number of seconds to wait after the end of a session for x6530 to close the window (to allow time for reading any messages that might appear if an error terminates the session). The default is zero. A value of -1 causes an indefinite wait
cursorColor (Foreground)	Specifies the text cursor color. The default is black.
dialog.background (Background)	Specifies the background color to be used for the informational dialog boxes (About x6530, etc.) The default is gray.
dialog.foreground (Foreground)	Specifies the foreground color to be used for the informational dialog boxes. The default is black.
dim (Dim)	Specifies the dim video color. The default is red.
font (Font)	Specifies the name of the normal font. The default is 9x15.
fontList (FontList)	Specifies the font used for menus and dialog boxes. The default is 9x15.
foreground (Foreground)	Specifies the color for displaying text in the window. Setting the class name instead of the resource name is an easy way to make everything that would normally appear in the text color change color. The default is black.
geometry (Geometry)	Specifies the preferred size and position of the window.
host (Host)	Specifies the name of the Tandem host to which x6530 is to be connected. There is no default for this option.
iconGeometry (IconGeometry)	Specifies the preferred size and position of the application when turned into an icon. It is not necessarily obeyed by all window managers.
iconName (IconName)	Specifies the icon name when x6530 is converted to an icon. The default is x6530.
jumpScroll (JumpScroll)	Turns the jumpscroll feature on or off. The default is on.
keymap (Keymap)	Specifies the keymap file to use. The default is the file .x6530modkey in the user's home directory or in the /etc directory. See "Keymap Files," later in this section, for details.

Table 2-2. Resource and Class Variables (Page 2 of 2)

Resource (Class)	Description
marginBell (MarginBell)	Specifies whether a bell rings when you type near the right end of a line. nMarginBell or Column defines the place on the line where the bell rings. The default is off.
menu.background (Background)	Specifies the background color to be used for the pop-up and cascading menus. The default is gray.
menu.foreground (Foreground)	Specifies the foreground color to be used for the pop-up and cascading menus. The default is black.
nMarginBell (Column)	Specifies the number of characters from the right margin at which the margin bell rings or the screen flashes.
pointerColor (Foreground)	Specifies the pointer color. The default is black.
pointerShape (Cursor)	Specifies the name of the shape for the pointer (mouse) cursor. Two values are available: xterm, which produces a pointer shape that looks like the letter "I," and arrow; the default is xterm. The pointer shape can be specified only in the .Xdefaults file.
port (Port)	Specifies the port number of the Telserv connection. The default is the default port set up for your system.
saveLines (SaveLines)	Specifies the number of lines to save beyond the top of the screen. The default is 480.
screenType (ScreenType)	Specifies the screen size that x6530 uses when in block mode. One of four numeric values can be specified: screenType 0 is the normal 24 lines by 80 columns; type 1 is 27x132, type 4 is 48x80, and type 5 is 54x132. The default is zero. (Types 4 and 5 are supplied especially for users of TEDIT.)
scrollbar.background (Background)	Specifies the color to be used for the scrollbar arrows and slider. The default is the same as the x6530 background color.
scrollbar.troughColor (TroughColor)	Specifies the color of the scrollbar region. The default is calculated from the background color to provide a 3-D effect.
title (Title)	Specifies a window title used by a window manager. This title is used only when a window manager places a title banner around a window. The default name is the value of the iconName resource.
visualBell (VisualBell)	Specifies whether to cause the window to flash when you type near the right margin. MarginBell or marginBell must also be set on for the visual bell option to work. nMarginBell or Column defines the place on the line where the screen flashes. The default for visualBell and VisualBell is off.

Note You can use wild cards to specify foreground and background colors for all aspects of x6530. For example, the following .Xdefaults file entries would set all background colors to blue and all foreground colors to white:

```
x6530*background: blue
x6530*foreground: white
```

**Reading Tandem
Commands From
a File**

When the `-exec` or `-dexec` command line option is used, x6530 initially looks for Tandem commands in a file, rather than receiving them from a keyboard. If you press any key on the keyboard while x6530 is reading the `-exec` or `-dexec` file, x6530 stops processing the file immediately and awaits further input from the keyboard.

The file contains commands in a sense-and-respond format as follows:

```
sense-string respond-string
```

sense-string

is one or more characters of a Tandem prompt with no embedded blanks. You do not need to specify the entire prompt, but only enough characters to identify it. For example, a `>` character is sufficient for the TACL prompt.

respond-string

is the character string to be sent to the Tandem host in response to *sense-string*. This string can contain embedded blanks.

The x6530 Emulator processes the lines in a command file sequentially; it compares a line's sense string to the prompt received from the Tandem host. When the sense string matches the Tandem prompt, x6530 sends whatever remains on the command line to the Tandem host.

Example

Suppose a command file contained the following lines:

```
Choice> tacl  
> logon hardware. engr  
Password: engr  
> files
```

When x6530 begins execution, it processes the above command file as follows:

1. The sense string in line 1, `Choice>`, matches the first prompt received from the Tandem host, `Enter Choice>`. The x6530 Emulator sends the Tandem host the respond string `tacl` so TACL is ready to receive the next commands.
2. The sense string in line 2, `>`, matches the TACL prompt sent from the Tandem host after Step 1. The x6530 Emulator sends the Tandem host the respond string consisting of a Tandem command (`logon`) and a logon ID (`hardware. engr`).
3. The sense string in line 3, `Password: ,` matches the prompt from the Tandem host after Step 2. The x6530 Emulator sends the host the respond string `engr`, the user's password. After the password is validated, the user is logged on to the Tandem host.
4. The sense string in line 4, another `>`, matches the prompt sent from the Tandem host after Step 3. The x6530 Emulator returns the respond string `files`, a TACL

command that lists the file names in the user's default volume on the Tandem host. After `files` has been executed, the user is left at the TACL prompt.

When x6530 is finished with the command file, it proceeds interactively. When the `-exec` option is used, the command file is kept after x6530 reads it. When the `-dexec` option is used, the command file is deleted after it is read.

x6530 Custom Commands Menu

The Custom Commands menu is created to run Guardian 90 programs. The instructions for this menu are put in a file named `.x6530rc` that must be in a user's home directory on the workstation where x6530 resides.

After `.x6530rc` is created, the Custom Commands menu is ready to use. The use of the menu is described in Section 1, "Using the x6530 Emulator."

The `.x6530rc` file contains options in the following format:

```
"menu-item" [ > TACL`command
               ! Guardian`90`program ]
```

`"menu-item"`

is the word or words listed as an item in the custom menu. The `menu-item` must be surrounded by double quotation marks.

> `TACL command`

specifies a TACL command to execute. Any TACL command, with its appropriate options, can be put here. Anything displayed by the command (unless it is an explicit or implicit RUN command with the WINDOW option) appears in the x6530 window from which it was executed. A user must have a TACL prompt in the x6530 window when selecting this type of menu item from the custom menu.

! `Guardian`90`program`

specifies a Guardian 90 program to execute. This is similar to the preceding option except that a new x6530 window is created to display the output of the program. This option requires two additional lines in the `.x6530rc` file so another x6530 window can be started:

```
USER      logon-ID
PASSWORD  password
```

The `logon-ID` and `password` are information needed to log on to the Tandem host.

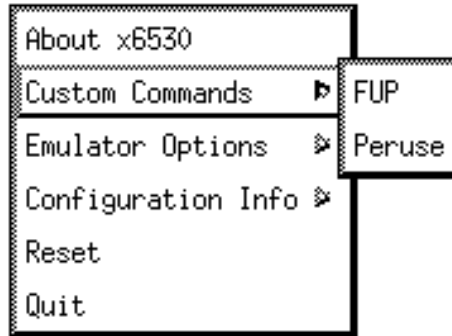
Example

The following example of a .x6530rc file shows the two types of custom menu items:

```
USER             hardware . engr
PASSWORD        sos
"FUP"           ! fup
"Peruse"        > peruse
```

Tandem logon information (the first three file entries) is included because FUP is set up to be displayed in a separate x6530 window.

The Custom Commands menu created from the above file would appear on the screen as follows:



013

Note The "!" form of the custom commands option starts a new x6530 session, logs on to TACL, then executes the requested program. To avoid starting an additional TACL process, it is better to use the ">" form of the custom commands option, specifying the WINDOW option. For example, such an entry in the .x6530rc file would appear as:

```
"FUP"    > FUP /WINDOW "-title FUP"/
```

Character Class for Copying and Pasting Text

When selecting text by words for a copy and paste operation, x6530 selects the characters making up the word based on a character class definition.

The x6530 Emulator uses the character classes shown in Table 2-3 as the default classes. These classes are for American English. Notice that some default classes have one character in them, while others have several, such as class 1. A character cannot be in more than one class.

Table 2-3. x6530 Default Character Classes (Page 1 of 2)

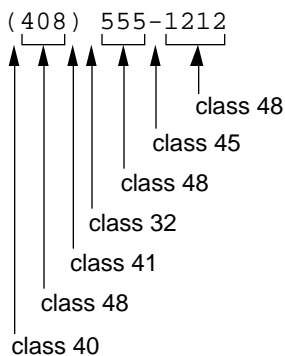
Class	Characters in Class									
1	ACK	BEL	BS	CAN	CR	DEL	DC1	DC2	DC3	
	DC4	DLE	EM	ENQ	EOT	ESC	ETB	ETX	FF	
	FS	GS	LF	NAK	RS	SI	SO	SOH	STX	
	SYN	SUB	US	VT						
32	HT	NUL	SP							
33	!									
34	"									
35	#									
36	\$									
37	%									
38	&									
39	'									
40	(
41)									
42	*									
43	+									
44	,									
45	-									
46	.									
47	/									
48	0	1	2	3	4	5	6	7	8	9
	A	B	C	D	E	F	G	H	I	J
	K	L	M	N	O	P	Q	R	S	T
	U	V	W	X	Y	Z	a	b	c	d
	e	f	g	h	i	j	k	l	m	n
	o	p	q	r	s	t	u	v	w	x
	y	z	_ (underscore)							

Table 2-3. x6530 Default Character Classes (Page 2 of 2)

Class	Characters in Class
58	:
59	;
60	<
61	=
62	>
63	?
64	@
91	[
92	\
93]
94	^
96	`
123	{
124	
125	}
126	~

How x6530 Selects a Word The x6530 Emulator follows these rules to determine which characters make up the word that is selected to copy:

- The character the pointer is positioned on in the word determines the character class to use for comparison purposes during the selection process. For example, the following telephone number represents 5 default character classes:



014

If the pointer were positioned on any of the numbers, class 48 would be the class to compare against. If the pointer were positioned on the hyphen, class 45 would be used, and so on.

- The x6530 Emulator scans the characters on either side of the pointer, comparing each one to the character class of the pointer character. When a character matches the class, it is selected, and the scanning continues. When a character does not match the class, the comparison process ends.
- The characters between the characters that stop the comparison process are selected as the word to be copied.

Examples

The following examples assume that x6530 is using the default character classes.

1. Assume that you have positioned the pointer on a word as follows:

The quick brown fox jumps over the lazy dog.

015

The pointer is on the lowercase letter `p`, which is in class 48. When you double-click the left mouse button, x6530 scans to the left of `p`, finding three characters, `jum`, that are in class 48; it stops when it encounters the blank character (class 32) in front of the `j`. It also scans to the right of `p`, finding one more character, `s`, that is in class 48. The blank character after the `s` stops the selection process in that direction. For this example, x6530 selects the word `jumps` to copy.

2. Assume that you have positioned the pointer on a word as follows:

Call this number: (415) 555-3962

016

The pointer is on the number `5`, which is in class 48. When you double-click the left mouse button, x6530 scans to the left of `5`, stopping when it encounters the `(` character. It also scans to the right of `5` until it finds the `)` character. The parentheses are not in class 48, so they act as stopping points for the selection process. For this example, x6530 selects `415` to copy.

3. Using the telephone number example again, assume the pointer is positioned as follows:

Call this number: (415) 555-3962

017

The pointer is positioned on the `)` character, which is in class 41. When you double-click the left mouse button, x6530 scans to the left of `)`, stopping when it finds the number `5`, a character that is not in class 41; it also scans to the right of `)` where it finds the blank character, another character not in class 41. The only character selected to copy is the `)` character.

Changing Character Class Definitions You can change the characters in the default character classes or create new character classes by these methods:

- Using the x6530 command line option `-cc` at run time.
- Using a `charClass` resource variable in the `.Xdefaults` file in the user's home directory on the workstation where x6530 resides.
- Using the `CharClass` class variable in the `.Xdefaults` file in the user's home directory on the workstation where x6530 resides.

The format for these methods is:

```
[ -cc
  charClass ] char-code:class [ , char-code:class ... ]
  CharClass ]
```

char-code

is the character code that represents the character you want in the class. The code is a number in the range of 0 through 255. A complete list of character codes and their corresponding characters is shown in Table 2-4. The character set Tandem supports is the ISO 8859.1 (International Standards Organization) character set—also known as the ECMA-94 (European Computer Manufacturers Association) Latin Alphabet Number 1.

You can have more than one character in a class; specify the code and class as many times as you need to define the characters. If you accidentally specify the same code more than once, only the last one encountered is used.

If the characters are represented by adjacent character codes in Table 2-4, you can specify the beginning and ending character codes rather than specify each code. The format is:

```
beginning-ending:class
```

The value of *beginning* must be less than the value of *ending*.

class

is a number of your choice to identify the character class; it can be a default class number or a new class number.

Table 2-4. Tandem Character Set (Page 1 of 7)

Code	Character	Character Name/Meaning
0	NUL	Null
1	SOH	Start of heading
2	STX	Start of text
3	ETX	End of text
4	EOT	End of transmission
5	ENQ	Enquiry
6	ACK	Acknowledge
7	BEL	Bell
8	BS	Backspace
9	HT	Horizontal tabulation
10	LF	Line feed
11	VT	Vertical tabulation
12	FF	Form feed
13	CR	Carriage return
14	SO	Shift out
15	SI	Shift in
16	DLE	Data link escape
17	DC1	Device control 1
18	DC2	Device control 2
19	DC3	Device control 3
20	DC4	Device control 4
21	NAK	Negative acknowledge
22	SYN	Synchronous idle
23	ETB	End of transmission block
24	CAN	Cancel
25	EM	End of medium
26	SUB	Substitute
27	ESC	Escape
28	FS	File separator
29	GS	Group separator
30	RS	Record separator
31	US	Unit separator
32	SP	Space
33	!	Exclamation point
34	"	Quotation mark
35	#	Number sign

Table 2-4. Tandem Character Set (Page 2 of 7)

Code	Character	Character Name/Meaning
36	\$	Dollar sign
37	%	Percent sign
38	&	Ampersand
39	'	Apostrophe
40	(Opening parenthesis
41)	Closing parenthesis
42	*	Asterisk
43	+	Plus
44	,	Comma
45	-	Hyphen (minus)
46	.	Period
47	/	Slash
48	0	Zero
49	1	One
50	2	Two
51	3	Three
52	4	Four
53	5	Five
54	6	Six
55	7	Seven
56	8	Eight
57	9	Nine
58	:	Colon
59	;	Semicolon
60	<	Less than
61	=	Equal to
62	>	Greater than
63	?	Question mark
64	@	Commercial at sign
65	A	Uppercase A
66	B	Uppercase B
67	C	Uppercase C
68	D	Uppercase D
69	E	Uppercase E

Table 2-4. Tandem Character Set (Page 3 of 7)

Code	Character	Character Name/Meaning
70	F	Uppercase F
71	G	Uppercase G
72	H	Uppercase H
73	I	Uppercase I
74	J	Uppercase J
75	K	Uppercase K
76	L	Uppercase L
77	M	Uppercase M
78	N	Uppercase N
79	O	Uppercase O
80	P	Uppercase P
81	Q	Uppercase Q
82	R	Uppercase R
83	S	Uppercase S
84	T	Uppercase T
85	U	Uppercase U
86	V	Uppercase V
87	W	Uppercase W
88	X	Uppercase X
89	Y	Uppercase Y
90	Z	Uppercase Z
91	[Opening bracket
92	\	Backslash
93]	Closing bracket
94	^	Circumflex
95	_	Underscore
96	`	Grave accent
97	a	Lowercase a
98	b	Lowercase b
99	c	Lowercase c
100	d	Lowercase d
101	e	Lowercase e
102	f	Lowercase f
103	g	Lowercase g

Table 2-4. Tandem Character Set (Page 4 of 7)

Code	Character	Character Name/Meaning
104	h	Lowercase h
105	i	Lowercase i
106	j	Lowercase j
107	k	Lowercase k
108	l	Lowercase l
109	m	Lowercase m
110	n	Lowercase n
111	o	Lowercase o
112	p	Lowercase p
113	q	Lowercase q
114	r	Lowercase r
115	s	Lowercase s
116	t	Lowercase t
117	u	Lowercase u
118	v	Lowercase v
119	w	Lowercase w
120	x	Lowercase x
121	y	Lowercase y
122	z	Lowercase z
123	{	Opening brace
124		Vertical line
125	}	Closing brace
126	~	Tilde
127	DEL	Delete
128–159		<i>not used</i>
160	NBS	No-break space
161	¡	Inverted exclamation mark
162	¢	Cent sign
163	£	Pound sign
164	¤	Currency sign
165	¥	Yen sign
166		Broken bar
167	§	Paragraph sign
168	¨	Diaeresis
169	©	Copyright sign

Table 2-4. Tandem Character Set (Page 5 of 7)

Code	Character	Character Name/Meaning
170	ª	Feminine ordinal indicator
171	«	Left angle quotation mark
172	¬	NOT sign
173	SH	Soft hyphen
174	®	Registered trade mark sign
175	–	Macron
176	°	Degree sign/ring above
177	±	Plus-minus sign
178	²	Superscript two
179	³	Superscript three
180	´	Acute accent
181	µ	Micro sign
182	¶	Pilcrow sign
183	•	Middle dot
184	¸	Cedilla
185	¹	Superscript one
186	º	Masculine ordinal indicator
187	»	Right angle quotation mark
188	¼	Vulgar fraction one quarter
189	½	Vulgar fraction one half
190	¾	Vulgar fraction three quarters
191	¿	Inverted question mark
192	À	Capital letter A with grave accent
193	Á	Capital letter A with acute accent
194	Â	Capital letter A with circumflex accent
195	Ã	Capital letter A with tilde
196	Ä	Capital letter A with diaeresis
197	Å	Capital letter A with ring above
198	Æ	Capital diphthong A with E
199	Ç	Capital letter C with cedilla
200	È	Capital letter E with grave accent
201	É	Capital letter E with acute accent
202	Ê	Capital letter E with circumflex accent
203	Ë	Capital letter E with diaeresis

Table 2-4. Tandem Character Set (Page 6 of 7)

Code	Character	Character Name/Meaning
204	Ì	Capital letter I with grave accent
205	Í	Capital letter I with acute accent
206	Î	Capital letter I with circumflex accent
207	Ï	Capital letter I with diaeresis
208	Ð	Capital Icelandic letter eth
209	Ñ	Capital letter N with tilde
210	Ò	Capital letter O with grave accent
211	Ó	Capital letter O with acute accent
212	Ô	Capital letter O with circumflex accent
213	Ö	Capital letter O with diaeresis
214	Õ	Capital letter O with tilde
215	×	Multiplication sign
216	Ø	Capital letter O with oblique stroke
217	Ù	Capital letter U with grave accent
218	Ú	Capital letter U with acute accent
219	Û	Capital letter U with circumflex
220	Ü	Capital letter U with diaeresis
221	Ý	Capital letter Y with acute accent
222	Þ	Capital Icelandic letter thorn
223	ß	Small German letter sharp s
224	à	Small letter a with grave accent
225	á	Small letter a with acute accent
226	â	Small letter a with circumflex accent
227	ã	Small letter a with tilde
228	ä	Small letter a with diaeresis
229	å	Small letter a with ring above
230	æ	Small diphthong a with e
231	ç	Small letter c with cedilla
232	è	Small letter e with grave accent
233	é	Small letter e with acute accent
234	ê	Small letter e with circumflex accent
235	ë	Small letter e with diaeresis

Table 2-4. Tandem Character Set (Page 7 of 7)

Code	Character	Character Name/Meaning
236	î	Small letter i with grave accent
237	í	Small letter i with acute accent
238	î	Small letter i with circumflex
239	ï	Small letter i with diaeresis
240	ð	Small Icelandic letter eth
241	ñ	Small letter n with tilde
242	ò	Small letter o with grave accent
243	ó	Small letter o with acute accent
244	ô	Small letter o with circumflex accent
245	õ	Small letter o with tilde
246	ö	Small letter o with diaeresis
247	÷	Division sign
248	ø	Small letter o with oblique stroke
249	ù	Small letter u with grave accent
250	ú	Small letter u with acute accent
251	û	Small letter u with circumflex accent
252	ü	Small letter u with diaeresis
253	ý	Small letter y with acute accent
254	þ	Small Icelandic letter thorn
255	ÿ	Small letter y with diaeresis

Examples

1. Suppose you want to add the following characters to the default class 48:

Circumflex (character code 94).

Period (character code 46).

Opening and closing brackets (character codes 91 and 93).

Backslash (character code 92).

Either of these methods would add the characters to class 48:

- a) Use the x6530 command line option `-cc` at run-time:

```
x6530 madcap -display fb:0 -cc 94:48,46:48,91-93:48
```

- b) Use the `charClass` resource variable in the `.Xdefaults` file in the user's home directory:

```
x6530.*.charClass: 94:48,46:48,91-93:48
```

2. Suppose you want to create three classes, as follows:

- A class containing the lowercase letters a-z and uppercase letters A-Z.
- A class containing the numbers 0-9.
- A class containing only the underscore character.

One way to create these classes is to use the following command line options:

```
x6530 s44 -display fb:0 -cc 65-90:300,97-122:300,48-57:400 &
```

This example creates class 300 for the lowercase and uppercase letters and class 400 for the numbers. Reassigning the letters and numbers from class 48 to new classes leaves only one character in class 48, the underscore character.

Using National Languages With x6530

In addition to American English, x6530 supports the alphabets and special characters of other spoken languages. To use this feature, you must install the language font file or files each user needs on their workstation (see the installation notes in the Tandem file named INSTALL, included with the x6530 software).

A language is selected with a standard X Toolkit command line option, `-fn`, and remains in effect until the x6530 session ends. A user cannot switch to another language during a session even though they may have more than one language font at their workstation.

All national language fonts are fixed-width fonts, displayed in one size only (6x13).

Table 2-5 lists the supported languages and the corresponding option values to use with `-fn`.

Table 2-5. National Languages for x6530

Language	-fn Option value
Danish	danish
English (United Kingdom)	english
Finnish	swedfin
French (AZERTY)	french
German	german
Norwegian	norwegian
Spanish	spanish
Swedish	swedfin

For example, the following command directs x6530 to use the Norwegian character set instead of the American English character set:

```
x6530 euro -display frank:0 -fn norwegian &
```

Keymap Files The x6530 Emulator can assign any key or any combination of keys on a workstation keyboard to a 6530 terminal key or function by using the information in a keymap file. Keymap files reside on the workstation that x6530 runs on, not the workstation where the x6530 window displays its output.

Keymap files depend on the X server and keyboard of a workstation. Several sample keymap files are provided with x6530. One of the sample keymap files, with or without modification, can be used as the default keymap file. Other sample files can be used as alternative keymap files to the default file if your users have a variety of keyboards and X servers. Alternative keymap files are defined with a `-keymap` command line option when x6530 is run.

When a `-keymap` option is not used, x6530 searches for the default keymap file named `.x6530modkey`, first in the user's home directory, then in the `/etc` directory. If the default file is not found in either place, x6530 stops because a keymap file is essential for it to run.

See the installation notes in the file named `INSTALL`, included with your x6530 software, for more information about the sample keymap files. You can print `INSTALL` at the Tandem host with a command similar to this one:

```
9> TGAL /IN $SYSTEM.ZX6530.INSTALL, OUT $$.#PTR1/
```

Example

The user is running x6530 at a UNIX workstation where the default keymap file is set up for the UNIX workstation. The user's computer is a Macintosh IIX named `mac3`, the workstation where the x6530 window will display its output. The x6530 run command would include a `-keymap` command line option to specify a keymap file for the Macintosh keyboard and X server:

```
x6530 sos -display mac3:1 -keymap /etc/.xkmMac
```

x6530 searches for `/etc/.xkmMac` on the UNIX workstation.

Creating a New Keymap File You need to create a new keymap file when the supplied keymap files do not support your user's keyboard and X server. Before creating the new keymap file, you must know the key symbol names (also called keysyms) for the keys on the keyboard. Determining all the keysyms is a lengthy process, but you must be thorough in discovering what the keysyms are for a particular keyboard.

Keysyms depend on the X server. For example, assume you have a PC that runs UNIX as its primary operating system and DOS as its secondary operating system. Under the UNIX X server, the keysyms for certain keys are one thing and under the DOS X server the keysyms for the same keys might be something else.

Figure 2-2 shows a partial keymap file for a Sun keyboard. The lines beginning with a `#` character are comments that show the keysym names for the Sun keyboard keys. The lines without a `#` character are the keymap instructions that assign the Sun keysyms to 6530 terminal keys and functions.

Figure 2-2. Sample Keymap File (partial listing)

```

# Key Cap on the Sun Keyboards
#
# L1 - L10
# F1 - F9
# Esc
# Tab
# Control
# Shift (left side)
# Shift (right side)
# Caps
# Left
# Right
# Alternate
# Linefeed
# Delete
# BackSpace
# R1 - R7
# R9
# R11
# R13
# R15
# R8
# R8
# R10
# R10
# R12
# R12
# R14
# R14
# A - Z
# a - z
# 0 - 9
# Return

Key Symbol
F11-F20 or L1 - L10
F1 - F9
Escape
Tab
Control_L
Shift_L
Right_R
Caps_Lock
Meta_L
Meta_R
Break
Linefeed
Delete
BackSpace
F21-F27 or R1 - R7
F29 or R9
F31 or R11
F33 or R13
F35 or R15
Up (unshifted)
F28 or R8 (shifted)
Left (unshifted)
F30 or R10 (shifted)
Right (unshifted)
F32 or R12 (shifted)
Down (unshifted)
F34 or R14 (shifted)
A - Z
a - z
0 - 9
Return

F1 F11
F2 F12
F3 F13
F4 F14
F5 F15
F6 F16
F7 F17
F8 F18
F9 F19
F10 F20
F11 Control_L F11
F12 Control_L F12
F13 Control_L F13
F14 Control_L F14
F15 Control_L F15
F16 Control_L F16

SF1 Shift_L F11
SF2 Shift_L F12
    
```

Comments

Keymap Instructions

Here is one way to create a new keymap file and the steps to follow:

1. Copy an existing sample keymap file from the Tandem host or the UNIX workstation to the user's home directory and rename it ".x6530modkey." The sample keymap file gives you a template of all the 6530 terminal functions to define.
2. Run the `xev` client to determine the keysym or keysyms for the keys on the keyboard (see the *X Window System User's Guide* for `xev` run instructions).

`xev` displays several pieces of event information in the `xterm` window after you press a key; you want the keysym value. For example, assume you have an IBM-compatible PC with a 101-key keyboard and the PC-Xview X server. When you press the 9 key, `xev` displays the events shown in Figure 2-3. The keysym for the 9 key is 9.

Figure 2-3. Keysym for 9 Key

```
KeyPress event, serial 13, synthetic NO, window 0x700001,
  root 0x8006b, subw 0x700002, time 64784128, (31,51), root:(480,80),
  state 0x0, keycode 17 (keysym 0x39, 9), same_screen YES,
  XLookupString gives 1 characters: "9"
```

019

If you pressed the Shift key and the 9 key on the same workstation, `xev` displays two events, one for the Shift key and one for the 9 key (see Figure 2-4). The keysyms for pressing the Shift and 9 keys are `Shift_L` and `parenleft`.

Figure 2-4. Keysyms for Shift and 9 Keys

```
KeyPress event, serial 15, synthetic NO, window 0x700001,
  root 0x8006b, subw 0x700002, time 64786432, (31,51), root:(480,80),
  state 0x0, keycode 49 (keysym 0xffe1, Shift_L), same_screen YES,
  XLookupString gives 0 characters: ""
```

```
KeyPress event, serial 15, synthetic NO, window 0x700001,
  root 0x8006b, subw 0x700002, time 64787008, (31,51), root:(480,80),
  state 0x1, keycode 17 (keysym 0x28, parenleft), same_screen YES,
  XLookupString gives 1 characters: "("
```

020

3. Write down the keysyms for all the keys on the keyboard and combinations of keys. "Key combination" means pressing a key with a modifier key such as the Shift key. Your X server documentation can tell you which modifier keys will generate different keysyms. This knowledge will save you time because you will not have to check every key combination on the keyboard.

4. Update all the comments at the beginning of the keymap file. You should be able to look at these comments and find any key on the keyboard and its keysym.
5. Each line of a keymap file defines one 6530 terminal key or function. A 6530 terminal key or function can have more than one definition. For example, the following two lines assign two keysyms on a keyboard to the 6530 BACKSPACE key:

```
BACKSPACE BackSpace
BACKSPACE Delete
```

When either key representing the keysyms BackSpace and Delete is pressed, the cursor moves backwards from its current position.

Build the keymap file using the keysyms. The format for a line is:

```
6530-key [ modifier-keysym ... ] main-keysym
```

6530-key

is the label printed on a 6530 terminal key (A, B, C..., 0, 1, 2, 3..., F1, Home, and so on), or a name representing a combination of keys that make up a function, such as CLEARABS (Shift/Ctrl Tab). Usually, you will be mainly interested in the 6530 functions that x6530 supports as shown in Table 2-6. All sample keymap files contain a complete list of 6530 functions. For more information on the 6530 functions, see the *653X Multi-Page Terminal Installation and Operation Guide*.

modifier-keysym

is one or more keysyms used with *main-keysym*. Shift, Ctrl, or Meta are usually named, but other keysyms can be used.

main-keysym

is the primary keysym assigned to *6530-key*.

Examples

1. To assign the L1 key on a Sun keyboard (keysym = F11) to the 6530 F1 function key, the following line is required in the keymap file:


```
F1 F11
```
2. Sometimes your keyboard does not physically have the keys you need. You can use the keymap file to assign a key or keys from your keyboard to a 6530 key or function. For example, suppose you have a PC with an 84-key keyboard that does not have the function keys F11, F12, and F13.

To assign PC keyboard keys to the 6530 function keys, the following keysyms are assigned in the keymap file:

```
F11 Control_L F1
F12 Control_L F2
F13 Control_L F3
```

In this example, when a user presses the Ctrl key and the F1, F2, or F3 key, x6530 interprets that as F11, F12, or F13, respectively.

Table 2-6 describes the functions available on a 6530 terminal.

Table 2-6. 6530 Keyboard Functions (Page 1 of 2)

6530 Function	Label on Key of 6530 Keyboard	Meaning
BACKTABS	Shift/Back Tab	Tab backward
BACKSPACE	Backspace	Move cursor to left
BREAK	Break	Break character
CLEARALLTABS	Ctrl/Shift/Clr Tab	Clear all tabs
CLEARTABS	Shift/Clr Tab	Clear a tab
DELETECHAR	Char Del	Delete a character
DELETELINE	Shift/Del Line	Delete a line
DNARROW	↓	Move cursor down
END	Ctrl/Home	Move cursor to position after last data item on screen
ENDOFLINE	Ctrl/Return	Move cursor to end of data on screen
ENDOFFPAGE	Shift/Home	Move cursor to end of display memory
ERASEENDOFLINE	Ctrl/Erase Line	Erase to end of line
ERASEENDOFFPAGE	Ctrl/Shift/Erase Page	Erase to end of page
FIRSTPAGE	Shift/Next Page	Next page (*)
F1	F1	Unshifted F1
:	:	:
:	:	:
F16	F16	Unshifted F16
HOME	Home	Move cursor to beginning of display memory
INSERTCHAR	Char Ins	Insert a character
INSERTMODE	Shift/Char Ins	Insert mode
INSERTLINE	Ins Line	Insert a line
LASTPAGE	Shift/Prev Page	Previous page (*)
LTARROW	←	Move cursor left
PGDN	Prev Page	Previous page
PGUP	Next Page	Next page

(*) This function depends on the Tandem application.

Table 2-6. 6530 KeyboardFunctions (Page 2 of 2)

6530 Function	Label on Key of 6530 Keyboard	Meaning
ROLLDN	Roll Down	Roll down
ROLLUP	Roll Up	Roll up
RTARROW	→	Move cursor right
SETTABS	Tab Set	Set a tab
SF1	Shift/F1	Shifted F1
⋮	⋮	⋮
SF16	Shift/F16	Shifted F16
SHIFTEDROLLDN	Shift/Roll Down	Roll down (*)
SHIFTEDROLLUP	Shift/Roll Up	Roll up (*)
STARTOFLINE	Shift/Return	Move cursor to beginning of line
TABS	Tab	Horizontal tab
UPARROW	↑	Move cursor up

Functions Added to x6530

6530 Function	Label on Key of 6530 Keyboard	Meaning
NEXTWORD	—	Move cursor to next word on line
PREVWORD	—	Move cursor to previous word on line

6530 Functions Not Supported by x6530

6530 Function	Label on Key of 6530 Keyboard	Meaning
	Print	
	Esc	
	Shift/Esc	
	Ctrl/Next Page	

(*) This function depends on the Tandem application.

Creating a New Keymap File for a National Language

The x6530 Emulator supports the ISO/ECMA international character set. This enables you to express multiple languages. To use this feature, make sure that x6530 is configured to use one of the ISO/ECMA fonts supplied with your workstation. Use the `-fn` command line option, or the font resource, to do so.

Formerly, the only way x6530 could express national language characters was by redefining a small subset of the ASCII character set. Although the full international character set is now available, the earlier method can still be used. In this method, the ASCII characters in the first row of Table 2-7 are redefined by x6530 to be displayed on the screen as the characters shown below them.

Table 2-7. National Characters Supported by x6530

Language	Characters											
ASCII (U.S.)	#	\$	@	[\]	^	`	{		}	~
French (AZERTY)	£	\$	à	°	ç	§	^	`	é	ù	è	¨
German	#	\$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
Spanish	#	\$	@	i	Ñ	¿	°	´	{	ñ	}	~
English (U.K.)	£	\$	@	[\]	^	`	{		}	~
Swedish/Finnish	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Danish	#	\$	@	Æ	Ø	Å	^	`	æ	ø	å	ü
Norwegian	£	\$	@	Æ	Ø	Å	^	`	æ	ø	å	ü

Displaying a national character on a screen works through a combination of hardware and software actions. Assume you have the following:

- A Danish keyboard with most of the Danish characters from Table 2-7 printed on some of the keys.
- The x6530 Danish language font file stored on your workstation.
- x6530 running with the `-fn` option specifying the Danish language font file.

When you press the Å key with the Shift key, x6530 receives the] character, searches the Danish language font file based on the] character, and displays the font character in that position of the font file, the uppercase Å character.

A problem arises when the character displayed on the screen is not the same as the character printed on the key you pressed. You can solve this problem by modifying a keymap file and making that file the default keymap file.

To make the keys display what is printed on them, follow these steps to change the default keymap or an alternate keymap file:

1. Run the `xev` client (see the *X Window System User's Guide* for `xev` run instructions). Using `xev`, press each key on your keyboard with and without the modifier keys to determine what each key displays and what its keysym is.
2. Note the keysyms for each key and key combinations involving modifier keys.

3. Update the comments at the beginning of the keymap file so you have a complete list of keyboard keys and their keysyms.
4. Add keymap instructions to the keymap file to assign the keysyms for the keys you need to redefine to the appropriate ASCII characters from Table 2-7 (#, \$, @, [, \,], ^, ` , {, |, }, or ~).

Example

Using the Danish language for this example, suppose that when you press the keys printed with Æ, Ø, and Å, with and without the Shift key, none of those characters are displayed on your screen. Instead, the characters :, ", {, ;, ', and [are displayed. You want the Æ, Ø, and Å keys to display the characters printed on them in uppercase and lowercase letters.

After running `xev` to find all the keysyms for your keyboard and X server, you know that the keys printed with Æ, Ø, and Å have the keysyms listed in Table 2-8.

Table 2-8. Danish Keysyms

Key	Keysym	What is Displayed	What You Want Displayed
Shift	Shift_L colon	:	Uppercase
Shift	Shift_L quotedbl	"	Uppercase
Shift	Shift_L braceleft	{	Uppercase
Æ	semicolon	;	Lowercase
Ø	quoteright	'	Lowercase
Å	bracketleft	[Lowercase

Looking in Table 2-7, you see that x6530 displays the national characters you want based on receiving the ASCII characters [, \,], {, |, and }. Table 2-9 summarizes this information.

Table 2-9. x6530 ASCII Characters for Your Keys

Key	What You Want Displayed	ASCII Character
Shift Æ	Uppercase Æ	[
Shift Ø	Uppercase Ø	\
Shift Å	Uppercase Å]
Æ	Lowercase æ	{
Ø	Lowercase ø	
Å	Lowercase å	}

To have the characters printed on the keys displayed when you press them, you add the following lines to the keysym file, assigning the keysyms listed in Table 2-8 to the ASCII characters from Table 2-9:

```
[  Shift_L colon
\  Shift_L quotedbl
]  Shift_L braceleft
{  semicolon
|  quoteright
}  bracketleft
```

Now, for example, when you press the Æ key with the Shift key, the Shift_L colon keysyms are sent to x6530; however, x6530 interprets these keysyms as the [character because of the first of the preceding keymap instructions. Then x6530 searches the Danish language font file based on the [character to find the uppercase Æ character to display on the screen.

Executing the x6530 Run Command

Most, perhaps all, users of x6530 will not have x6530 residing at their workstations; this is also true for a window manager. The more likely scenario is that x6530 and a window manager will reside at some other host in the network. A user, however, will probably want a window manager active at their workstation while they use x6530 and other clients because the window manager provides many useful services.

With x6530 and the window manager residing on a workstation other than their own, how does a user conveniently issue run commands for these clients from their workstation? There are three approaches you can use in this situation:

- Set up the X server at a user's workstation to automatically start a window manager.
- Use a Telnet X client built into the user's X terminal to start a window manager.
- Set up xdm to automatically start the X server and a window manager.

Each approach permits the user to run x6530 from the window manager's menu, assuming the window manager's menu can be customized to run user-defined clients. Each approach is discussed below using mwm as the example window manager and assumes that mwm resides on the same UNIX workstation as x6530.

Setting Up the UNIX Workstation

Each of the approaches presented in the previous discussion requires you to do some preparatory work for each user on the UNIX workstation where x6530 resides. The steps are as follows:

1. Create an account for each user.
2. Include a path command and UNIX environment DISPLAY variable in each user's UNIX .login file.
3. Place resource or class variables for x6530 in a file named .Xdefaults in each user's home directory (an .Xdefaults file is optional).

4. Place customizing information for mwm in a file named `.mwmrc` in each user's home directory.

User Accounts and Path Commands Use the method for your UNIX system to establish a user account for each x6530 user. After creating the user accounts, each user's UNIX `.login` file should have a path command that includes the directory where the X Window System files are stored, usually `/usr/bin/X11`. For example:

```
path = (/bin /user/bin/ /etc /usr/bin/X11)
```

DISPLAY Variable The UNIX `.login` file is a convenient place to define the user's workstation as the place to display the x6530 window. This specification is done with a UNIX environment `DISPLAY` variable. For example:

```
setenv DISPLAY steve:0
```

Note Another way to define the display workstation is with the `x6530 -display` option.

.Xdefaults File Each user can customize the way x6530 works with an `.Xdefaults` file. To start, make a copy of the sample `.Xdefaults` file that was included with the X Window System in each user's home directory. You or the user can edit this file to add or change the resource and class variables for x6530.

Setting Up mwm's Menu To run x6530 from mwm's menu, the menu must be customized to include an item for x6530 for the user to select. The menu item is defined in the file named `.mwmrc` in the user's home directory.

A sample `.mwmrc` file is provided with the X Window System. Make a copy in each user's home directory that either you or the user can modify to include an x6530 run command as an mwm menu item. See the *X Window System User's Guide* for more information about customizing and using mwm.

For example, the following portion of a `.mwmrc` file defines four menu items, including one named `x6530`. When `x6530` is selected from mwm's menu, the `x6530` command it is associated with is executed:

```
menu RootMenu
{
  "x6530"          f.exec "x6530 T44 -keymap /etc/.xkmPCXView &"
  "UNIX Window"   f.exec "xterm &"
  "Clock"         f.exec "xclock &"
  "Editor"        f.exec "xedit &"
  "Quit MWM"      f.quit_mwm
}
```

-
- Automatically Starting mwm** You can automatically start mwm by using:
- The workstation's X server, if the X server can start a client.
 - xdm, a standard X client.
 - A Telnet X client.
- Using an X Server to Start mwm** X servers that start clients include servers that run on DOS PCs and Macintosh computers. Some X servers offer this feature only at the beginning of an X session, while others offer it continually throughout the X session.
- PC-Xview is an X server from Graphic Software Systems, Inc., that operates under DOS. A configuration file, created when installing PC-Xview, would contain information to designate mwm as the first client to start and the name of the UNIX workstation to run it from. When PC-Xview is started, the user is prompted for his or her password on the UNIX workstation, then mwm is started immediately after the X server comes up at the workstation.
- X servers from Apple Computer, Inc., and White Pine Software, Inc., can start clients. For these X servers, an X client is started by selecting commands from a pull-down clients menu. The user adds commands to the clients menu through a dialog box. X clients can be added or modified at any time while the X server is running. In this case, the user could start mwm from the pull-down menu, then start x6530 from mwm's menu, or set up the clients menu to have separate commands for mwm and x6530.
- Using a Telnet X Client to Start mwm** The major X terminal vendors have Telnet X clients built into the setup screens for the X server. The user switches back and forth between the X server and set-up modes with a single keystroke. The user uses Telnet in setup mode to access the UNIX workstation to start mwm, then switches back to server mode to start x6530 from mwm's menu. The disadvantage of this method is that more UNIX experience is required of the user, such as knowing the UNIX commands to log on, change directories, run processes, stop processes, obtain directory listings, and so on. Some knowledge of Telnet is also required.
- Using xdm to Start mwm** xdm is a good choice to start mwm if you do not have X servers that start clients. xdm is a standard X client included with X that automatically starts one or more clients for a workstation from your instructions. After xdm is started, it runs continuously, polling each workstation to determine if it should start an X session and the defined clients, such as mwm, for that workstation.
- A user never has to think about starting an X server and mwm with xdm. All a user sees is a login window for the UNIX workstation whenever he or she turns on the workstation. After the user logs in to UNIX, mwm is started for the user's workstation; then x6530 can be run from mwm's menu. See the *X Window System User's Guide* for complete information on setting up and running xdm.

Index

6530 terminal keyboard functions 2-27

A

About x6530 menu option 1-6
Accessing the Tandem host 1-2
Automatically starting mwm 2-33
Autorepeat menu option 1-9
Autorepeat option 2-2, 2-6

B

Background color option 2-6
Block mode applications 1-4, 1-15, 1-16
Block of text, selecting to copy 1-17
Bold font option 2-2, 2-6
Border color option 2-6
Border width option 2-6

C

Character class option 2-2, 2-6, 2-11, 2-14
Character classes
 changing definitions 2-14
 codes 2-14
 defaults 2-11
 description of 2-11
 display 1-13
Character set 2-14
Class variables
 Autorepeat 2-6
 Background 2-6, 2-7
 BorderWidth 2-6
 CharClass 2-6
 CloseDelay 2-6
 Column 2-7
 Cursor 2-7
 description of 2-5
 Dim 2-6
 Font 2-6
 FontList 2-6
 Foreground 2-6, 2-7
 Geometry 2-6

Class variables (continued)

Host 2-6
IconGeometry 2-6
IconName 2-6
JumpScroll 2-6
Keymap 2-6
MarginBell 2-7
Port 2-7
SaveLines 2-7
ScreenType 2-7
Title 2-7
VisualBell 2-7

Command file format 2-8

Command line options 2-1

-auto 2-2
-cc 2-2, 2-11, 2-14
-closeDelay 2-2
-cr 2-2
-dexec 2-2
-dim 2-2
-exec 2-2
-fb 2-2
-fn 2-2, 2-22
-help 2-2
-host 2-2
-iconic 2-2
-j 2-2
-keymap 2-3, 2-23
-mb 2-3
-ms 2-3
-n 2-3
-nb 2-3
-port 2-3
-screenType 2-3
-sl 2-3
-T 2-3
-vb 2-3
-xrm 2-3

Commands, reading from a file 2-8

Configuration Info menu 1-10

Configuration info menu option 1-6

Copying/pasting text
 copying from protected fields 1-18
 copying from unprotected fields 1-17
 pasting 1-18
 selecting text by block 1-17
 selecting text by line 1-17
 selecting text by page 1-17
 selecting text by word 1-17, 2-12
Creating a keymap file 2-23
Creating a national keymap file 2-29
Cursor color option 2-2, 2-6
Cursors 1-3, 1-4
Custom Commands menu 1-7, 2-9
Custom commands menu option 1-6
Customizing x6530
 character class 2-11, 2-14
 class variables 2-5
 command line options 2-1, 2-4
 national languages 2-22
 resource variables 2-5
 summary 2-1
 XENVIRONMENT variable 2-5
 .mwmrc file 2-32
 .x6530rc file 2-9
 .Xdefaults file 2-5, 2-32

D

Dexec file option 2-2
Dexec option 2-8
Dialog background option 2-6
Dialog foreground option 2-6
Dim option 2-2, 2-6
Display character classes option 1-13
Display keymaps option 1-12
Display resources menu option 1-11
DISPLAY variable 2-32
Display workstation 2-32

E

Emulator Options menu 1-8
Emulator options menu option 1-6
Entering characters in x6530 window 1-4
Event information for a keyboard 2-25
Exec option 2-2, 2-8
Executing x6530 2-1, 2-31

F

Fields, moving within 1-4
Foreground option 2-6
Function keys, 6530 terminal 1-16

G

Geometry option 2-6
Guardian 90 programs, executing 2-9

H

Help option 2-2
Host option 2-2, 2-6
HOSTS file, TCP/IP 2-1

I

Icon geometry option 2-6
Icon name option 2-3, 2-6
Icon startup option 2-2
Icon, turning window into 1-2
Installing keymap files 2-23

J

Jump scroll menu option 1-9
Jump scroll option 2-2, 2-6

K

Key symbols 2-23

Keyboard, event information for 2-25

Keymap files

- 6530 terminal functions in 2-27

- creating 2-23

- default 2-23

- description of 2-23

- for a national font 2-29

- installing 2-23

- instruction syntax 2-26

- keysyms 2-23

- modifier keysyms 2-26

- sample keymap file 2-23

- .x6530modkey 2-23

Keymap option 2-3, 2-6

Keymaps display 1-12

Keysyms 2-23

L

Language font files, national 2-22, 2-29

Line of text, selecting to copy 1-17

Logging on to Tandem host 1-2

Login file, UNIX 2-32

M

Margin bell menu option 1-9

Margin bell option 2-3, 2-7

Mark, positioning

- in protected fields 1-18

- in unprotected fields 1-18

Menu and dialog font option 2-6

Menu background option 2-7

Menu foreground option 2-7

Menus

- Configuration Info 1-6, 1-10
 - display character classes 1-13
 - display keymaps 1-12
 - display resources 1-11
- Custom Commands 1-6, 1-7, 2-9
- Emulator Options 1-6, 1-8
- main menu 1-5

Mouse

- activating x6530 menus with 1-5
- moving text cursor with 1-4
- simulating function keys with 1-16
- simulating paging keys with 1-15
- three-button 1-4

Multiple x6530 windows 1-2, 1-3

mwm

- customizing 2-32
- starting a client from 2-32
- starting automatically
 - with an X server 2-33
 - with xdm 2-33
- .mwmrc file 2-32

N

- National language files 2-22, 2-29
- nMargin bell option 2-7
- Normal font option 2-2, 2-6

P

- Page of text, selecting to copy 1-17
- Paging keys, 6530 terminal 1-15
- Pasting text in x6530 window 1-18
- Path command, UNIX 2-32
- PC-Xview 2-33
- Pointer color option 2-3, 2-7
- Pointer cursor 1-3, 1-4
- Pointer shape option 2-7
- Port option 2-3, 2-7
- Protected fields 1-4, 1-18

Q

Quit menu option 1-6

R

Reading commands from a file 2-8

Reset menu option 1-6

Resetting x6530 window 1-6

Resource variables

 autorepeat 2-6

 background 2-6

 boldFont 2-6

 borderColor 2-6

 borderWidth 2-6

 charClass 2-6

 closeDelay 2-6

 cursorColor 2-6

 description of 2-5

 dialog.background 2-6

 dialog.foreground 2-6

 dim 2-6

 font 2-6

 fontList 2-6

 foreground 2-6

 geometry 2-6

 host 2-6

 iconGeometry 2-6

 iconName 2-6

 jumpScroll 2-6

 keymap 2-6

 marginBell 2-7

 menu.background 2-7

 menu.foreground 2-7

 nMarginBell 2-7

 pointerColor 2-7

 pointerShape 2-7

 port 2-7

 saveLines 2-7

 screenType 2-7

 scrollbar.background 2-7

 scrollbar.troughColor 2-7

Resource variables (continued)

- title 2-7
- visualBell 2-7
- Resource-setting option 2-3
- Resources display 1-11
- Respond string 2-8
- Reverse video menu option 1-9
- Run command, x6530 2-1

S

- Save lines option 2-7
- Screen size option 2-3, 2-7
- Scroll output menu option 1-9
- Scrollbar background option 2-7
- Scrollbar option 2-3
- Scrollbar trough color option 2-7
- Scrollbar, description of 1-14
- Sense string 2-8
- Simulating function keys 1-16
- Simulating paging keys 1-15
- Sizing x6530 window 1-4, 1-9
- Starting mwm 2-33
- Stopping x6530 1-2, 1-6

T

TACL

- See* Logging on to Tandem host
- TACL command, executing 2-9

Tandem host

- logging on to 1-2
- name 2-1

TCP/IP

- HOSTS file 2-1
- network 2-1

Telnet X clients 2-33

Text cursor

- appearance of 1-3
- moving with keystrokes 1-4
- moving with mouse 1-4

Three-button mouse 1-4

Title option 2-7

U

- UNIX workstation
 - creating user accounts on 2-32
 - DISPLAY variable 2-32
 - path command 2-32
 - .login file 2-32
 - Unprotected fields 1-17
-

V

- Visual bell menu option 1-9
 - Visual bell option 2-3, 2-7
-

W

- WINDOW run option 1-2
 - Window size menu options 1-9
 - Window title option 2-3
 - Window, x6530
 - appearance of 1-1
 - entering characters in 1-4
 - erasing 1-6
 - multiple windows 1-2, 1-3
 - resetting 1-6
 - sizing of 1-4
 - turning into an icon 1-2
 - Window-closing delay option 2-2, 2-6
 - Word in text, selecting to copy 1-17
-

X

- X client
 - xdm 2-31, 2-33
 - xev 2-25
- X servers
 - for x terminals 2-33
 - information about 1-6
 - PC-Xview 2-33
- X terminals 2-33
- x6530 customizing files
 - .x6530rc 2-9
 - .Xdefaults 2-5, 2-32
- x6530 Emulator, description of 1-1

x6530 main menu 1-5
x6530 window
 See Window, x6530
xdm client 2-31, 2-33
XENVIRONMENT variable, UNIX 2-5
xev client 2-25

Special characters

-auto option 2-2
-cc option 2-2, 2-11, 2-14
-closeDelay option 2-2
-cr option 2-2
-dexec option 2-2
-dim option 2-2
-display option 2-32
-exec option 2-2
-fb option 2-2
-fn option 2-2, 2-22
-help option 2-2
-host option 2-2
-iconic option 2-2
-j option 2-2
-keymap option 2-3
-mb option 2-3
-ms option 2-3
-n option 2-3
-nb option 2-3
-port option 2-3
-screenType option 2-3
-sl option 2-3
-T option 2-3
-vb option 2-3
-xrm option 2-3
.mwmrc file 2-32
.x6530modkey file 2-23
.x6530rc file 1-7, 2-9
.Xdefaults file 2-5, 2-32