# X$^{Ten}$ User’s Guide

TENON
INTERSYSTEMS

*New Dimensions in Personal Workstation Technology*

1123 Chapala Street, Santa Barbara, CA 93101
TEL 805-963-6983 • FAX 805-962-8202
1-800-6-MACH-10
info@tenon.com • http: / / www.tenon.com

12/98

# TABLE OF CONTENTS

# 1.0 X$^{Ten}$ Overview

## 1.1 The X Window System

The X Window System®, or X, is a network-transparent graphics window system originally developed at the Massachusetts Institute of Technology. X is based on a "client / server" model, where an application program (the "client") communicates information to a display program (the "server"), which then outputs the information to a bitmapped screen. The server directs user input, typically from a keyboard or mouse, to the client program for interpretation.

Client programs are commonly run on the same machine as the server; however, clients may also run transparently from other machines connected to the server machine (sometimes called an "X terminal") via a network. The client and server communicate with one another using the X Protocol, a software communication protocol layered over TCP / IP.

X allows multiple clients to run simultaneously, each displayed in a separate, overlapping "window" on the server. Using the mouse and keyboard, the user controls the size, appearance and location of each window on the display, or "X desktop".

## 1.2 The X$^{Ten}$ Window System

X$^{Ten}$ transforms your Macintosh into a high-performance X workstation. It includes a full-featured, 24-bit, X11R6 X server and is the only Macintosh X Window software package that includes a local execution environment for X client applications. A suite of over one hundred X clients is included with X$^{Ten}$.

X$^{Ten}$ supports multiple, simultaneous threads of execution. With X$^{Ten}$, X client programs, as well as remote, network-based X clients, and local or remote X window managers, are integrated into a single Macintosh display. Simultaneous operation of Macintosh applications, local X clients and the X$^{Ten}$ X server makes your Power Macintosh a serious tool in distributed X Window environments.

# 2.0 Installing X$^{Ten}$

X$^{Ten}$ will run on all Power Macintoshes and Power Macintosh clones, with Macintosh Operating System (MacOS) version 7.1.2 or later.

We recommend configuring your Macintosh with at least 32MB of RAM. The X$^{Ten}$ installation consumes approximately 70MB of disk space.

Quit all currently running Macintosh applications before beginning the installation. AutoDoubler and any startup Macintosh applications should also be disabled. Holding down the <shift> key while rebooting will disable virus protection software and any startup routines that may interfere with the installation.

### *Step 1 — Insert the X$^{Ten}$ CD-ROM or Download the X$^{Ten}$ Installer*

### *Step 2 — Double-Click on the X$^{Ten}$ Installer and Follow the Instructions*

The following steps detail the installation instructions.

- The *XTen Installer* will create a folder called *XTen 6.1* on the hard disk you select. The “Basic” X$^{Ten}$ installation will require approximately 70MB of disk space. There are a number of installation variations, but they do not significantly alter the X$^{Ten}$ disk footprint.[1] The Basic installation does not include electronic online reference information or the complete set of X11R6 clients and fonts. Other installation variations include reference information, fonts, and the remaining set of local X11 client applications.

- The *XTen Installer* will also create a *Tenon Libraries* folder in the *XTen* folder. Do not move this folder.

- The *XTen Installer* will also install an X$^{Ten}$ control panel named *XTen Controls* in the *Control Panels* folder in your System Folder.

[1] Most of the X$^{Ten}$ files are stored in a single, large file named *XTen_FFS*. The Basic X$^{Ten}$ installation creates this file with sufficient extra room for the optional X$^{Ten}$ installation variations.

- The *XTen Installer* will also install files named *XTen Prefs* and *Tenon Icons* in the *Preferences* folder of your System Folder.

### Step 3 — Reboot the Macintosh

- You must reboot your system for the control panel to be recognized by the MacOS.

### Step 4 — Configure the X$^{Ten}$ Control Panel

- Before starting X, configure the X$^{Ten}$ Controls control panel according to the instructions in section "2.2 X$^{Ten}$ Control Panel".

## 2.1 X$^{Ten}$ Components

This release of the X$^{Ten}$ Window software is based on the public release of X11, Release 6, from MIT. The release contains an X Window server, fonts, a font server, and documentation from the X11R6 release. The release also includes a suite of X client applications for running locally on your X$^{Ten}$ server platform.

The main display server components provided are:

| | |
|---|---|
| XTen | The X$^{Ten}$ X Window server |
| afterstep | The AfterStep Window Manager (default) |
| olvwm | The OpenLook Virtual Window Manager and user environment |
| mwm | The Motif Window Manager and user environment |
| twm | The Tab Window Manager and user environment |
| xdm | X Display Manager |
| xinit | X server initialization program (default) |
| fonts | X11 fonts in Portable Compiled Format and a network font server |

After installation, the X11 software will reside in the following directories:

| | |
|---|---|
| XTen | X$^{Ten}$ Macintosh application |
| Tenon Libraries | X$^{Ten}$ PowerPC shared libraries |
| System Folder / Control Panels / XTen Controls | X$^{Ten}$ control panel |
| System Folder / Preferences / | X$^{Ten}$ icons and X$^{Ten}$ Prefs |

The following reside in *XTen_FFS*:

| | |
|---|---|
| / usr / X11R6 / lib / X11 | X11 configuration files, fonts and default resources |
| / usr / X11R6 / bin | X11 executable files |
| / usr / X11R6 / man / mann | X11 manual pages |

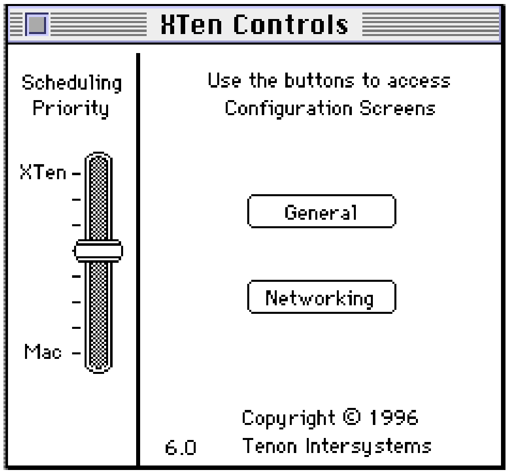
**IMPORTANT:** In the same manner that successfully booting and running the Macintosh OS is dependent on the System Folder and some very specifically named files and folders in the System Folder, X$^{Ten}$ is dependent on its file system hierarchy, including the names, locations and permissions of its files.

Be very careful when using Macintosh-imposed restrictions on the X$^{Ten}$ folders and files. This includes locking or assigning File Sharing privileges to any X$^{Ten}$ file or folder.

## 2.2 X$^{Ten}$ Control Panel

The X$^{Ten}$ Controls control panel is used to configure certain parameters in your X$^{Ten}$ system. It must be opened once prior to launching X$^{Ten}$ for the first time, and may be opened subsequently if you want to change X$^{Ten}$ operating parameters.

To view the X$^{Ten}$ Controls control panel, choose “Control Panels” from the Apple menu and double-click on “XTen Controls”. The control panel consists of the Scheduling Priority slide bar and two configuration buttons. With the exception of Scheduling Priority slide bar adjustments, all changes made in the X$^{Ten}$ Controls control panel require you to restart X$^{Ten}$ in order for the modifications to take effect.



Use the buttons to access Configuration Screens

Copyright © 1996
Tenon Intersystems

**Figure 1. X$^{Ten}$ Control Panel**

## 2.3 Scheduling Priority Slide Bar

On a busy system where both X$^{Ten}$ and Macintosh programs are running simultaneously, the Scheduling Priority slide bar allows you to control the relative CPU time spent running X$^{Ten}$ versus running Macintosh programs. When the Scheduling Priority slide bar is moved upward toward the "`XTen`" mark, X$^{Ten}$ will draw output faster to the screen. When the slide bar is moved downward toward the "`Mac`" mark, X$^{Ten}$ will surrender more time to other, simultaneously executing MacOS programs and Macintosh system functions such as keyboard input and mouse movement input events. The recommended slide bar setting that optimizes these potentially competing demands on X$^{Ten}$ is shown in "Figure 1. X$^{Ten}$ Control Panel". A change of priority goes into effect immediately, allowing you to tune your system depending on your personal workload priority. For optimal performance, the X$^{Ten}$ Controls control panel should be closed after making any scheduling priority adjustments.

## 2.4 Essential Configuration

The installation process requires knowledge of your network environment. You will need the following information:

- the local time zone
- the name of your host (the machine on which X$^{Ten}$ is being installed)
- your Internet address on Ethernet (or EtherTalk) and the subnetwork mask (if appropriate)
- your Internet address on AppleTalk and subnetwork mask (if appropriate)
- your domain name and the Internet address of your Domain Name Server
- the name and Internet address of your default gateway (you may be using either a MacIP gateway or a standard IP gateway)

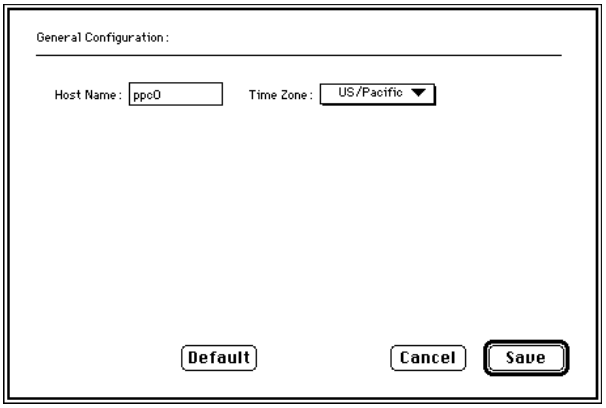
X$^{Ten}$ includes its own TCP protocol stack. If you have already configured the Apple TCP/IP control panel, X$^{Ten}$ will use the networking information provided by OpenTransport by default.

If you are using the OpenTransport option (see "Appendix G — Using OpenTransport or Replacing OpenTransport and OpenTransport"), X$^{Ten}$ will use the network information from your TCP / IP control panel and ignore the IP address settings in the X$^{Ten}$ control panel. X$^{Ten}$ will still use the Domain Name Server information and the "Enable Incoming Connections" check box from the X$^{Ten}$ control panel. For additional information on IP addresses, see "Appendix B — Internet (IP) Addresses".

☞ **Before starting X for the first time, you must open the X$^{Ten}$ Controls control panel and visit each configuration panel.**

Configuration options are selected by clicking an appropriate button in the X$^{Ten}$ control panel. Each screen contains edit fields and check boxes for specifying configuration values. The "Default" button in each configuration screen will reset the entries in the screen to installation default values. After entering your configuration values, click on the "Save" button to confirm your entry, or on the "Cancel" button to leave the values unchanged. Changed values will take effect the next time you start X$^{Ten}$.

## 2.4.1 General Configuration



General Configuration:

Host Name: ppc0

Time Zone: US/Pacific

Figure 2. General Configuration Screen

### 2.4.1.1 Host Name

This is the name given to your Macintosh. The host name entered will be used in conjunction with addressing information from the Networking Configuration screen (described below) to uniquely identify your system on an internet network. The default host name will be taken from the “Macintosh Name” field in the “Network Identity” section of the Sharing Setup control panel.

☞ **Under X$^{Ten}$, blanks and special characters other than periods and underscores are not allowed in a host name. Special characters found in the default host name are converted to underscores.**

### 2.4.1.2 Time Zone

This is a pop-up menu for choosing the appropriate time zone for your system. In order to integrate with time as seen on the Macintosh, X$^{Ten}$ uses the time as set in the Macintosh General control panel, typically local time. The time zone information helps X$^{Ten}$ convert between local time and GMT. Users are responsible for adjusting the General control panel clock by hand when changing to and from Daylight Savings Time.

If available, the default time zone is extracted from the setting in the Macintosh World Map control panel.

## 2.4.2 Networking Configuration Screen

X$^{Ten}$ allows for a multi-homed internet networking environment — permitting multiple independent networks to be connected to a single system. X$^{Ten}$ supports several physical interface types, including Ethernet, SLIP and LocalTalk, in addition to a logical local loopback interface. X$^{Ten}$ also supports multiple interfaces of the same type, including up to four Ethernets, three SLIP and / or PPP lines, and a single LocalTalk.

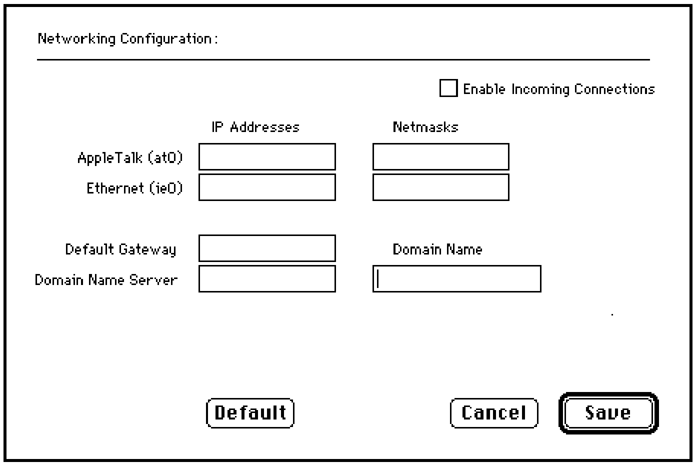


Figure 3. Networking Configuration Screen

### 2.4.2.1 Enable Incoming Connections

Check this box to accept connection requests for internet network services. When checked, X$^{Ten}$ will automatically start *inetd* — the internet "super server" — upon X$^{Ten}$ launch. *inetd* listens for connections on well-known internet "ports" such as Telnet and FTP. With *inetd* disabled, connection requests for these services will be denied. By default, *inetd* is disabled. Prior to enabling incoming connections, you should refer to "Appendix E — X$^{Ten}$ Security & Passwords".

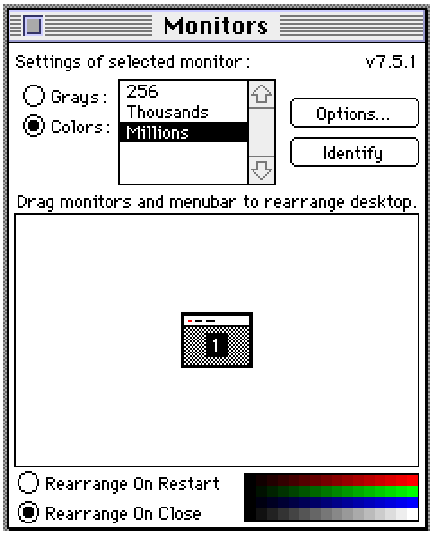
## 2.5 Preparing Your Macintosh Control Panels

When launched, X$^{Ten}$ will configure itself based on the settings in the Monitors, Sound, Mouse, and Keyboard control panels. X$^{Ten}$’s performance characteristics can be tuned in the X$^{Ten}$ Controls control panel.

You must visit the Monitors control panel prior to running X$^{Ten}$. Open your Monitors control panel and select from 256, Thousands or Millions in either the Grays or Colors settings.

In a multi-screen or video mirroring configuration, X$^{Ten}$ runs on the main screen (the screen containing the menu bar).

☞ **You are cautioned against changing the monitor characteristics while X$^{Ten}$ is running. Doing so will cause X$^{Ten}$ to restart automatically, using the new monitor characteristics.**



Monitors

Settings of selected monitor : v7.5.1

○ Grays :
● Colors :

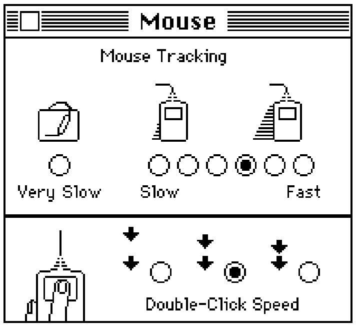
256
Thousands
Millions

Drag monitors and menubar to rearrange desktop.

○ Rearrange On Restart
● Rearrange On Close

**Figure 4. Monitors Control Panel**

Open your Mouse control panel and review your mouse tracking and double-click rate. X$^{Ten}$ obeys the settings in this control panel and ignores requests from X clients to change mouse characteristics.



Mouse

Mouse Tracking

○ ○○○●○○

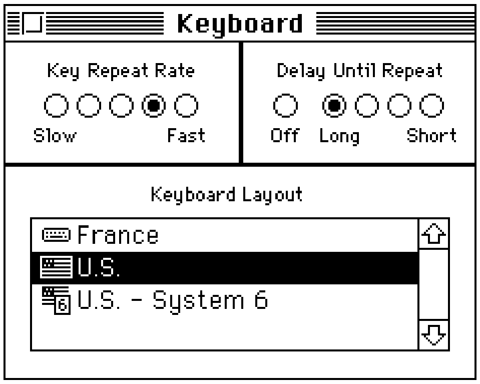
Very Slow Slow Fast

○ ● ○

Double-Click Speed

Figure 5.  Mouse Control Panel

Open your Keyboard control panel and review your keyboard repeat rate and delay, if any. X$^{Ten}$ obeys the settings in this control panel and ignores requests from X clients to change keyboard repeat frequency.



Keyboard

Key Repeat Rate

○○○●○

Slow Fast

Delay Until Repeat

○ ●○○○

Off Long Short

Keyboard Layout

France

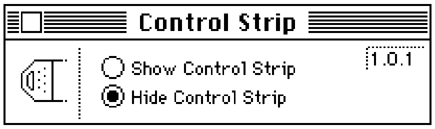
U.S.

U.S. - System 6

Figure 6.  Keyboard Control Panel

Open your Sound control panel and review the sound and intensity. X clients that ring the system bell will generate the selected sound. X$^{Ten}$ will also honor X client requests to change bell duration and to turn the bell on or off. These requests will not affect your Sound control panel settings.

If you have a Control Strip control panel, select the "Hide Control Strip" option when running X$^{Ten}$.

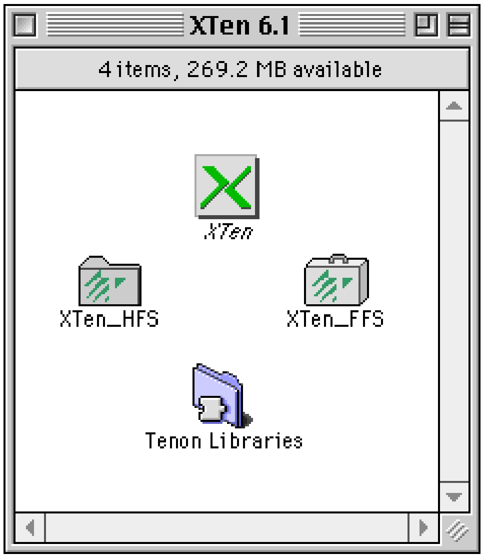


○ Show Control Strip
● Hide Control Strip

Figure 7.  Control Strip Control Panel

## 2.6 Launching X$^{Ten}$

Launch X$^{Ten}$ by clicking on the *XTen* application icon in the *XTen* folder. You may make an alias to this application and put it on the desktop for easier launching.

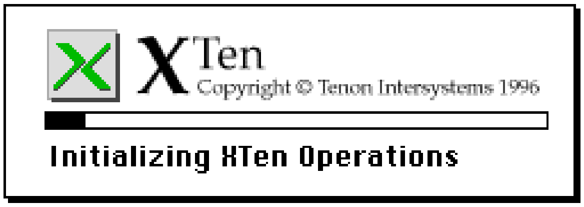


XTen 6.1

4 items, 269.2 MB available

**Figure 8. X$^{Ten}$ Folder**

The system will present a display with progress information on the startup and initialization process.



XTen

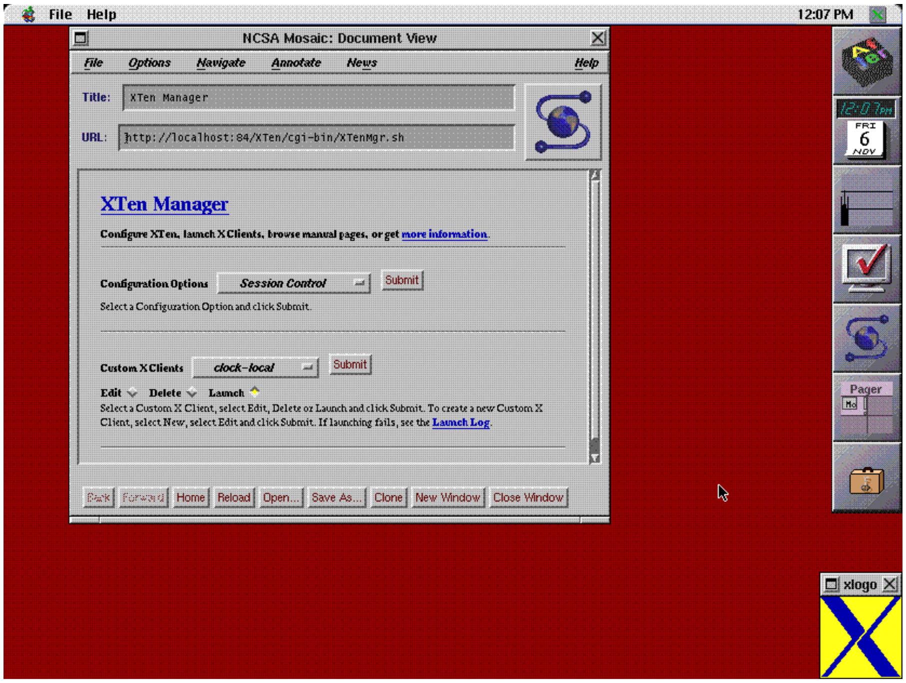
Copyright © Tenon Intersystems 1996

Initializing XTen Operations

**Figure 9. X$^{Ten}$ Progress Window**

The screen will darken briefly, and then change to a gray stippled pattern. This is the default state of the X11R6 X server. X$^{Ten}$ is configured to start an AfterStep Window Manager (*afterstep*). X$^{Ten}$ then starts the Mosaic Web browser with parameters to contact the local X$^{Ten}$ Manager for configuration, and to display the *xlogo* client.

The initial X$^{Ten}$ desktop is shown below.



Title: XTen Manager

URL: http://localhost:84/XTen/cgi-bin/XTenMgr.sh

**XTen Manager**

**Configure XTen, launch X Clients, browse manual pages, or get more information.**

**Configuration Options** *Session Control*

Select a Configuration Option and click Submit.

**Custom X Clients** *clock-local*

**Edit** ◇ **Delete** ◇ **Launch** ◆

Select a Custom X Client, select Edit, Delete or Launch and click Submit. To create a new Custom X Client, select New, select Edit and click Submit. If launching fails, see the **Launch Log**.

**Figure 10. X$^{Ten}$ AfterStep Desktop**

# 3.0 The X$^{Ten}$ Manager

The X$^{Ten}$ Manager is used to configure and maintain the X$^{Ten}$ application. Since the X$^{Ten}$ Manager is constructed with HTML (Hyper Text Markup Language), it can be accessed by any standard Web browser. This provides a clean, easy-to-use point-and-click user interface for configuring X$^{Ten}$.

The first time X$^{Ten}$ is run, the AfterStep Window Manager is started and the Mosaic Web browser is automatically launched, displaying the X$^{Ten}$ Manager’s configuration page. Since the X$^{Ten}$ Manager controls the automatic startup of the Mosaic browser, subsequent X$^{Ten}$ execution can be configured not to invoke the Mosaic browser. In addition, the window manager has a desktop menu which allows you to start the X$^{Ten}$ Manager on an “as needed” basis. See section “4.3 Navigating the OpenLook Desktop”.

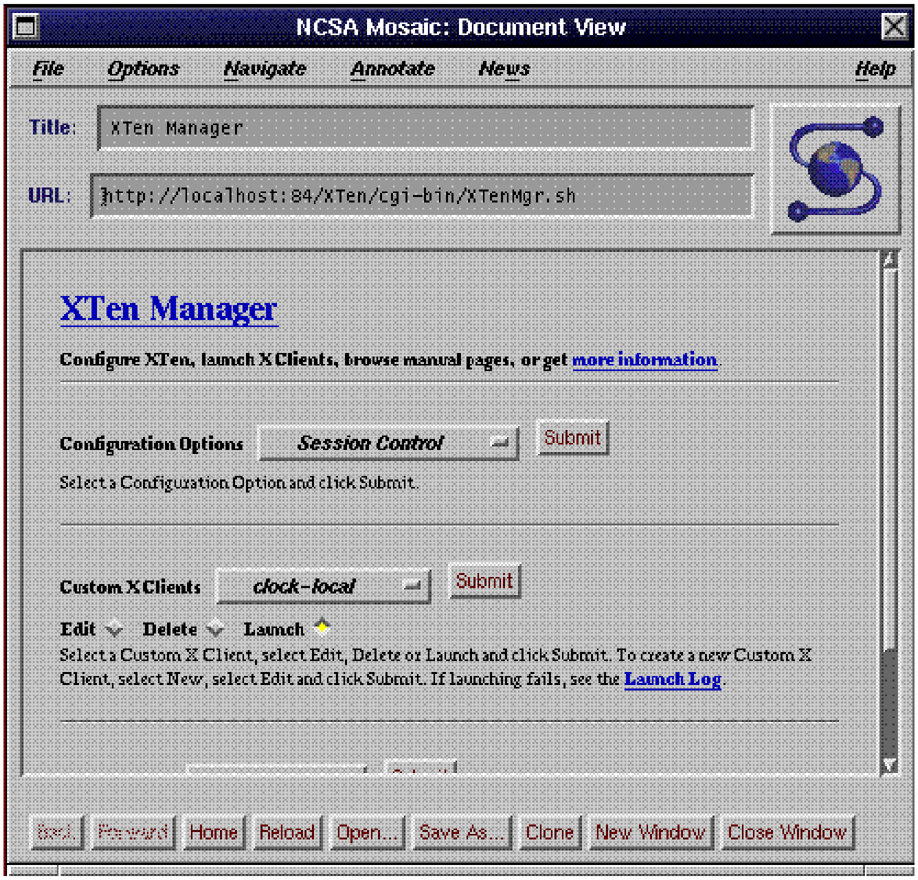
The X$^{Ten}$ Manager can be used to configure a variety of X characteristics, such as enabling the Macintosh menu bar and three-button mouse settings. The X$^{Ten}$ Manager can also be used to configure the automatic execution of both local X clients and scripts to access and start remote clients.

Access to the X$^{Ten}$ Manager is, at first, limited to a “local” Web browser. The X$^{Ten}$ application, however, may be reconfigured to support X$^{Ten}$ Manager access from a “remote” Web browser. Sophisticated host and network addressing is used to limit remote access to the X$^{Ten}$ Manager, but with proper configuration, a family of X$^{Ten}$ platforms can be managed from a single, central location using standard internet Web browser technology.

The X$^{Ten}$ Manager presents an X$^{Ten}$ configuration Web page with three or four constituent components:

- getting more information
- configuring X$^{Ten}$
- customizing X clients
- viewing the X$^{Ten}$ manual pages

Each component is separated by a Web delineation line. The “X$^{Ten}$ manual pages” section is displayed only if the “Optional Online Manuals” package has been installed.



Title: XTen Manager

URL: http://localhost:84/XTen/cgi-bin/XTenMgr.sh

XTen Manager

Configure XTen, launch X Clients, browse manual pages, or get more information.

Configuration Options: Session Control [Submit]

Select a Configuration Option and click Submit.

Custom X Clients: clock-local [Submit]

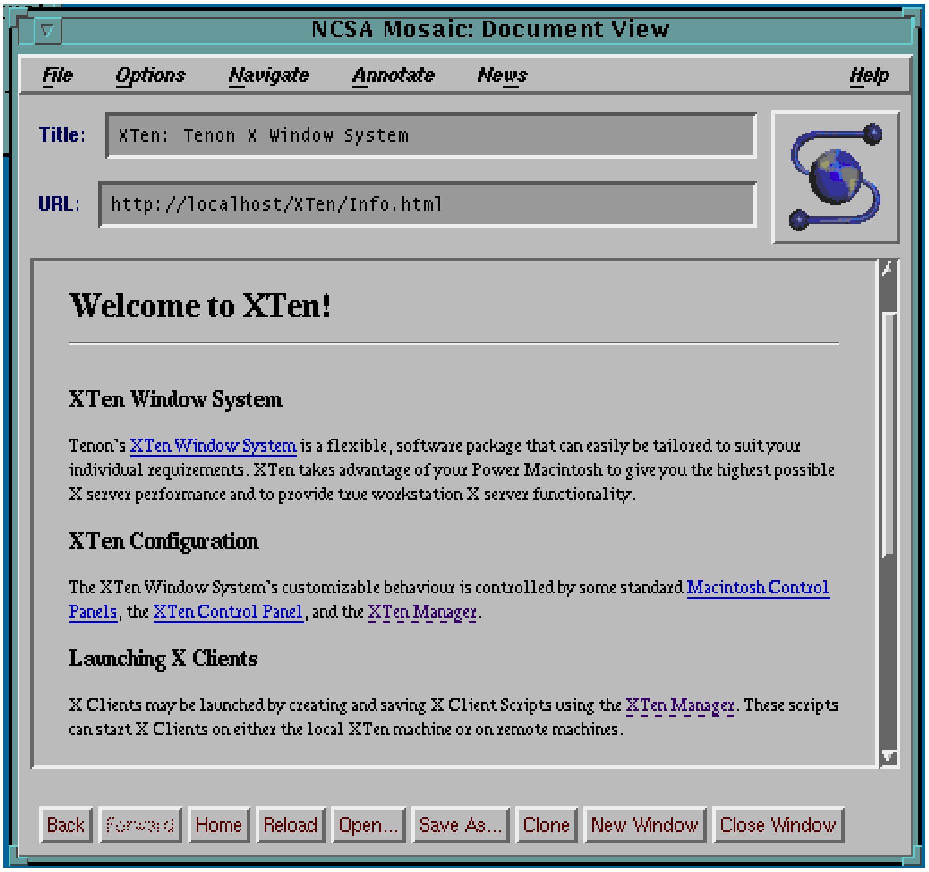
Edit ◇ Delete ◇ Launch ◆

Select a Custom X Client, select Edit, Delete or Launch and click Submit. To create a new Custom X Client, select New, select Edit and click Submit. If launching fails, see the Launch Log.

Figure 11. X$^{Ten}$ Manager

## 3.1 Getting More Information

The first component of the configuration page is the X$^{Ten}$ Manager information section. This section contains information about X$^{Ten}$ and an HTML hot link to Tenon's Home Page. It is worthwhile visiting this section to learn how to take full advantage of X$^{Ten}$ and how to access late-breaking information about X$^{Ten}$ over the Internet.



Title: XTen: Tenon X Window System

URL: http://localhost/XTen/Info.html

Welcome to XTen!

XTen Window System

Tenon's XTen Window System is a flexible, software package that can easily be tailored to suit your individual requirements. XTen takes advantage of your Power Macintosh to give you the highest possible X server performance and to provide true workstation X server functionality.

XTen Configuration

The XTen Window System's customizable behaviour is controlled by some standard Macintosh Control Panels, the XTen Control Panel, and the XTen Manager.

Launching X Clients

X Clients may be launched by creating and saving X Client Scripts using the XTen Manager. These scripts can start X Clients on either the local XTen machine or on remote machines.

**Figure 12. X$^{Ten}$ Manager Information Section**

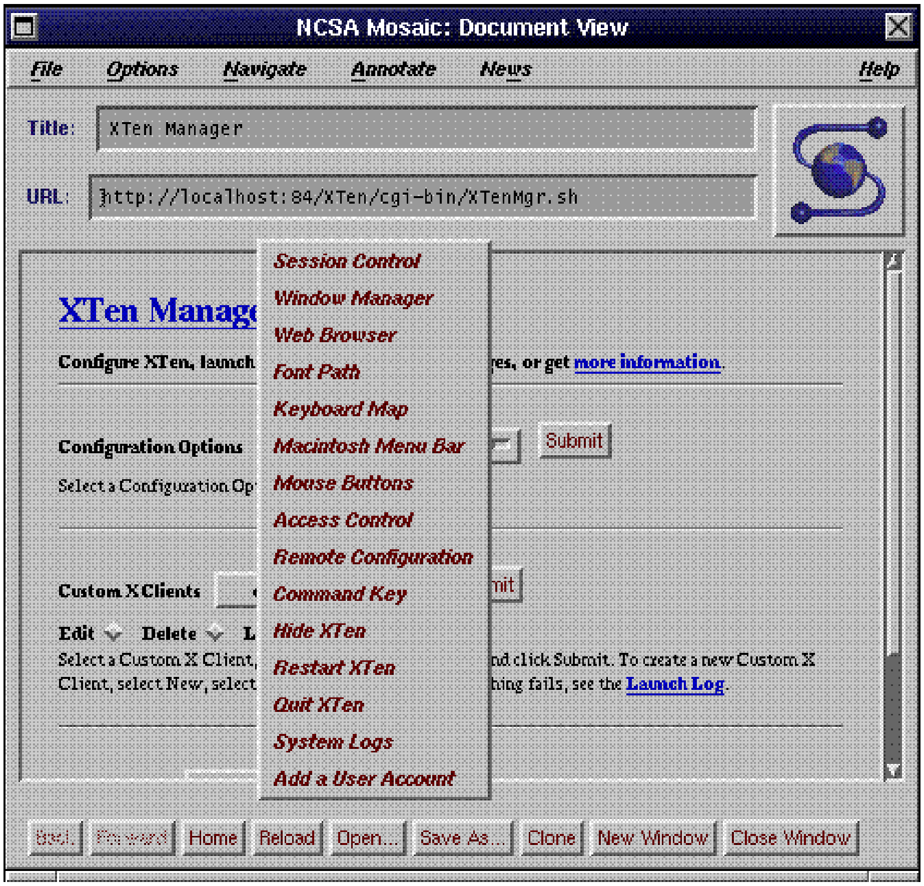
## 3.2 Configuring X$^{Ten}$

The second section of the configuration page contains configuration options. To change a part of the X$^{Ten}$ configuration, select an option with the mouse and click “Submit”. A new page will be displayed with information about the option to be configured and instructions on how to change the X$^{Ten}$ configuration.

Configuration options include:

- controlling XDM sessions
- changing the Web browser — controls the automatic startup of the X$^{Ten}$ Mosaic Web browser
- changing the window manager — changes the look and feel of the X desktop
- setting the font path — sets the font path to obtain X11 fonts either locally from a local X$^{Ten}$ font server or from a remote font server
- setting a new keyboard map
- enabling or disabling display of the Macintosh menu bar
- mapping mouse buttons
- setting access control to the local system from a remote system displaying X clients
- supporting access to the local system from a remote system for remote configuration
- controlling which X window keyboard modifier key is used when the Apple <Command> key is pressed
- hiding, restarting and quitting X$^{Ten}$
- displaying various system logs
- adding user accounts

Each of the configuration options is self-explanatory, and includes hyper text selections that you can click on to explain the configuration in more detail. When your configuration is complete, you can click on the "Back" button at the bottom of the Web page and return to the previous page.



Title: XTen Manager

URL: http://localhost:84/XTen/cgi-bin/XTenMgr.sh

XTen Manag

Configure XTen, launch … es, or get more information.

Configuration Options

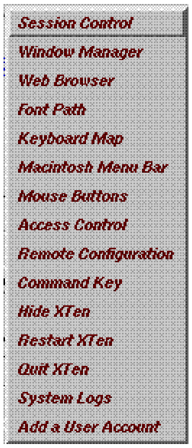
Select a Configuration Op

Custom X Clients

Select a Custom X Client, … nd click Submit. To create a new Custom X Client, select New, select … hing fails, see the Launch Log.

Figure 13. X$^{Ten}$ Manager Configuration Options

## 3.2.1 Controlling XDM Sessions



X$^{Ten}$ can be configured to control X sessions using either the *Xinit* program or XDM. *Xinit* provides the simplest level of control. It manages a single X session and doesn't require user authentication. XDM can manage a collection of either local or remote X displays. Under XDM, users are presented with a uniform entry into the X Window system. User authentication is required.

The "Xinit" and "Local XDM" settings do not require the specification of a remote host name in the "XDM Server" field. For "Remote XDM" or "Chooser XDM", the "XDM Server" field must be filled in with the host name of the XDM server. This server is the system that X$^{Ten}$ will contact when X$^{Ten}$ is started or new X sessions are established.

Select the appropriate form of session control and click "Save". The current X session is not automatically terminated, nor is the new session control immediately activated. When changing from "Xinit" (or any other XDM type) to an XDM session, it is only necessary to quit the existing X$^{Ten}$ session for the new changes to take effect. When changing from the local XDM setting to "Xinit", it is necessary to completely quit X$^{Ten}$ and relaunch X$^{Ten}$ for the new changes to take effect.

### 3.2.1.1 Accessing the X$^{Ten}$ Manager During a Remote XDM Session

Configuring X$^{Ten}$ to establish a remote XDM session transfers control of the X$^{Ten}$ desktop to the remote XDM system. The X$^{Ten}$ Manager can transfer desktop control back to the X$^{Ten}$ system, but access to the X$^{Ten}$ Manager may be restricted by the remote system. (This depends on the configuration of the session on the remote system but a typical configuration would default to not allowing such connections.)

A number of options are available for accessing the X$^{Ten}$ Manager during a remote XDM session. One of the solutions require customizing the X$^{Ten}$ Manager before establishing a remote XDM session, but the other solutions handle the "chicken and egg" case where remote XDM is established first.

Option #1

Use the Finder to launch a Macintosh Web browser (e.g., Netscape) directly on the X$^{Ten}$ system. Enter the following URL to access the X$^{Ten}$ Manager:

```
http://127.0.0.1:84/
http://localhost:84/
```

This solution assumes that you have enough memory to run a browser on the X$^{Ten}$ system and that the X$^{Ten}$ system is configured to use OpenTransport. (Using OpenTransport is the default X$^{Ten}$ configuration.)

Option #2

Using the remote system, enable access to the X$^{Ten}$ desktop from the X$^{Ten}$ system itself. An easy way to do this is to open an xterm on the remote system and issue an *xhost* command. Use the IP address of the X$^{Ten}$ system as the parameter to the *xhost* command.

```
# xhost <IP address of XTen>
```

Then open an xterm on the X$^{Ten}$ system. Pull down X$^{Ten}$'s File menu and select "xterm", or type <Command-Option-T> on the X$^{Ten}$ keyboard. When the local xterm comes up, enter the following command to launch the Mosaic browser with a connection to the X$^{Ten}$ Manager.

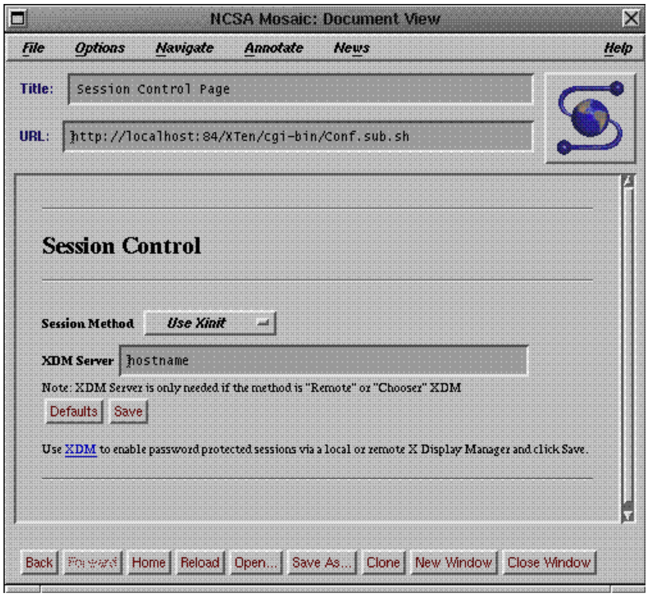
```
# ./.browser
```

Option #3

Use a Web browser on any other system (including the remote XDM system) and connect to the X$^{Ten}$ Manager on the X$^{Ten}$ system. This approach requires that the IP address of the remote system is added to X$^{Ten}$'s "Remote Configuration" (see section "3.2.9 Remote Configuration Permissions") settings before the desktop session control is transfered to a remote XDM server. If you have the foresight to added this setting before establishing a remote XDM session, the X$^{Ten}$ Manager may be contacted at any time thereafter.

Use the IP address or hostname of the X$^{Ten}$ system in the following URL to connect to the XTen Manager

```
http://<IP of XTen>:84/
http://hostname:84/
```



Title: Session Control Page

URL: http://localhost:84/XTen/cgi-bin/Conf.sub.sh

Session Control

Session Method: Use Xinit

XDM Server: hostname

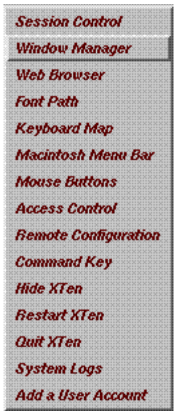
Note: XDM Server is only needed if the method is "Remote" or "Chooser" XDM

Defaults | Save

Use XDM to enable password protected sessions via a local or remote X Display Manager and click Save.

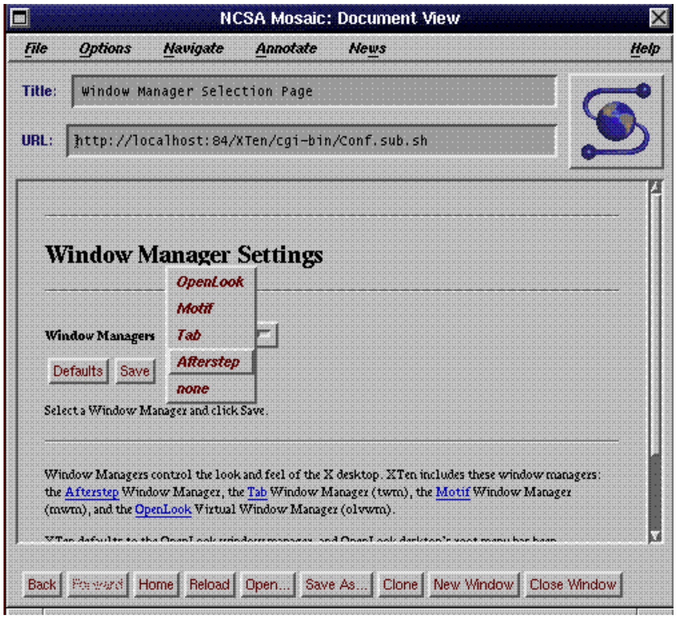
Figure 14. Session Control

## 3.2.2 Choosing a Window Manager



AfterStep, OpenLook, Motif and Toms Window Managers are shipped with X$^{Ten}$. Any one of these window managers may be selected, or you may choose "none" for no window manager. After a window manager is selected, you must click "Save" to notify the X$^{Ten}$ Manager of your choice.

Reference information for each of the window managers is available with "AfterStep", "OpenLook", "Motif" and "Toms" HTML hot link selections.



Title: Window Manager Selection Page

URL: http://localhost:84/XTen/cgi-bin/Conf.sub.sh

Window Manager Settings

Window Managers

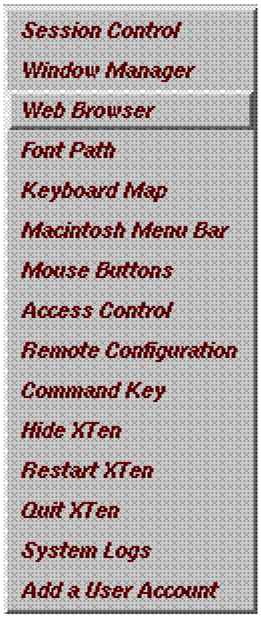
Select a Window Manager and click Save.

Window Managers control the look and feel of the X desktop. XTen includes these window managers: the Afterstep Window Manager, the Tab Window Manager (twm), the Motif Window Manager (mwm), and the OpenLook Virtual Window Manager (olvwm).

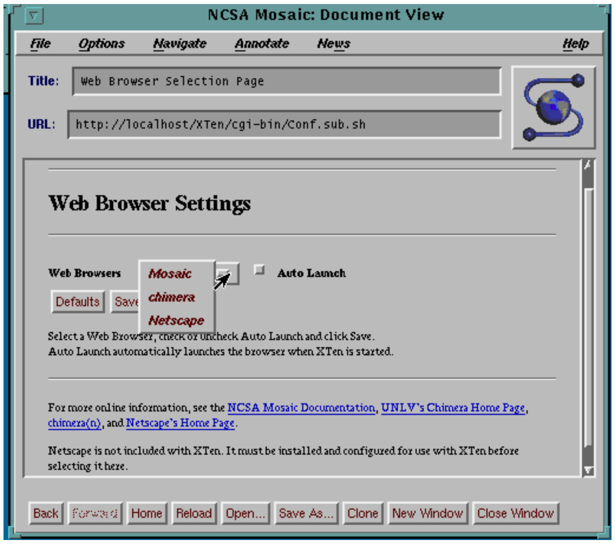
XTen defaults to the OpenLook window manager, and OpenLook desktop's root menu has been

Figure 15. Window Manager Settings

## 3.2.3 Choosing a Web Browser



Any Web browser may be used by X$^{Ten}$ to access the X$^{Ten}$ Manager. In addition to the Mosaic Web browser, the Netscape and Chimera Web browsers may be selected. Both Netscape and Chimera are available from the Internet and need to be obtained separately. The "Auto Launch" button controls whether the selected Web browser is automatically started when X$^{Ten}$ is started. The default configuration for the Mosaic Web browser is "Auto Launch". Reference information about each of the browsers is available by clicking on the "NCSA Mosaic Documentation", "UNLV's Chimera Home Page" and "Netscape's Home Page" HTML hot links. After selecting a Web browser and an Auto Launch setting, you must click on the "Save" button to activate the setting.



Title: Web Browser Selection Page

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

## Web Browser Settings

Web Browsers Mosaic □ Auto Launch

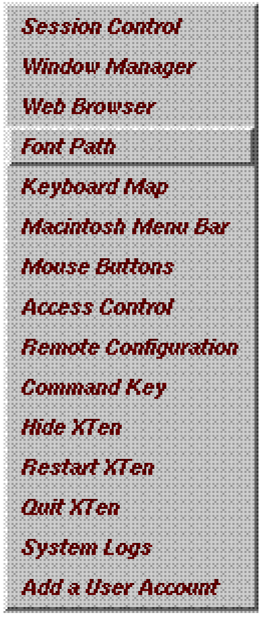
Select a Web Browser, check or uncheck Auto Launch and click Save.
Auto Launch automatically launches the browser when XTen is started.

For more online information, see the NCSA Mosaic Documentation, UNLV's Chimera Home Page, chimera(n), and Netscape's Home Page.

Netscape is not included with XTen. It must be installed and configured for use with XTen before selecting it here.

**Figure 16. Web Browser Settings**

## 3.2.4 Defining a Font Path



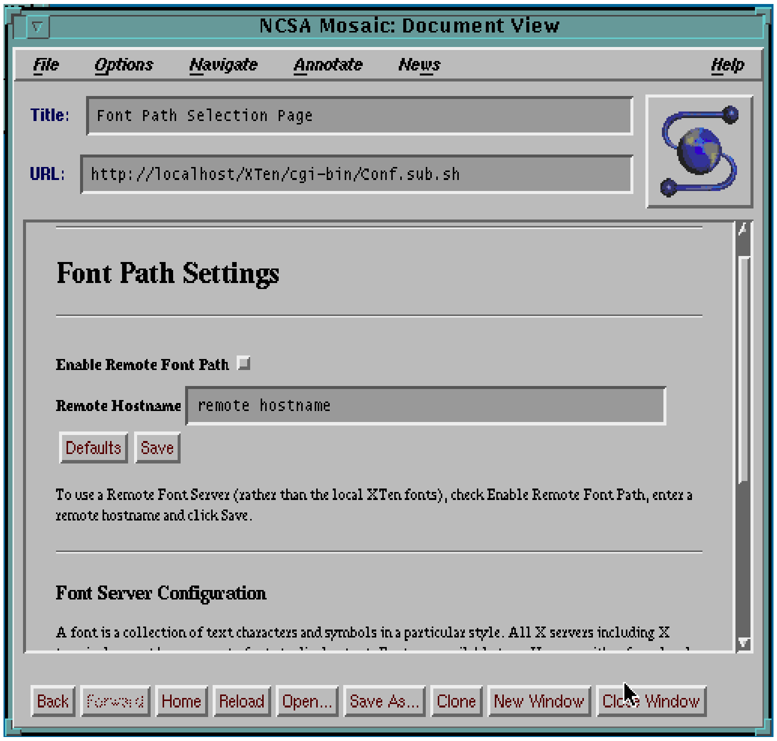
In order for any X server to display information, a set of fonts is required. Since storage for fonts can take a significant amount of space, X$^{Ten}$ has the ability to communicate with a separate entity known as a "font server". Thus, a single font server may be run within a networked cluster of X servers so that a single copy of the fonts may be shared across a network by all of the X servers. By default, X$^{Ten}$ uses a family of locally installed X11 fonts. This configuration item selects whether fonts are to be found locally or remotely.

Installation of the "Complete" X$^{Ten}$ software package enables X$^{Ten}$ to become a font server for network-based X terminals and servers. Refer to section "8.1 Providing Fonts Over the Network" for more information.

The "font detail" HTML hot link describes the local font path and fonts in more detail.

### 3.2.4.1 Obtaining Fonts from a Network Font Server

By default, X$^{Ten}$ will look for fonts from the local directories described in the previous section. Use the X$^{Ten}$ configuration page to obtain fonts from a network font server when X$^{Ten}$ starts up.



Title: Font Path Selection Page

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

**Font Path Settings**

**Enable Remote Font Path** □

**Remote Hostname** remote hostname

Defaults Save

To use a Remote Font Server (rather than the local XTen fonts), check Enable Remote Font Path, enter a remote hostname and click Save.

**Font Server Configuration**

A font is a collection of text characters and symbols in a particular style. All X servers including X

Figure 17. Font Path Settings

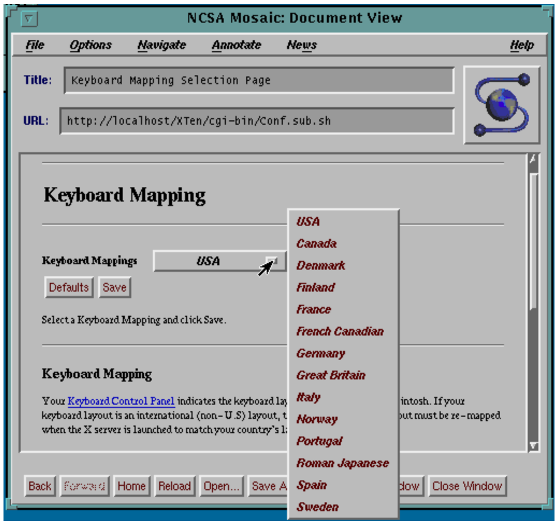
To specify a remote font server, the host name or Internet address of the computer that supports that server must be specified in this configuration item. Once the remote host information is given, the "Enable Remote Font Path" button must be selected. Once that is done, the "Save" button must be selected to inform the X$^{Ten}$ Manager of your choice.

## 3.2.5 Selecting a Keyboard Mapping

The Macintosh Keyboard control panel indicates a keyboard layout for a Macintosh. If a keyboard is not a U.S. layout, the keyboard layout must be re-mapped to correspond to the actual key settings.

This configuration item provides for the selection of default keyboard mappings from a variety of alternative non-U.S. keyboard layouts. Select a keyboard mapping that corresponds to your keyboard and click on the "Save" button to inform the X$^{\text{Ten}}$ Manager of your choice.

If your keyboard layout is not among the alternative choices, click on the "xmodmap" HTML hot link for more detailed information.



Title: Keyboard Mapping Selection Page

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

Keyboard Mapping

Keyboard Mappings: USA

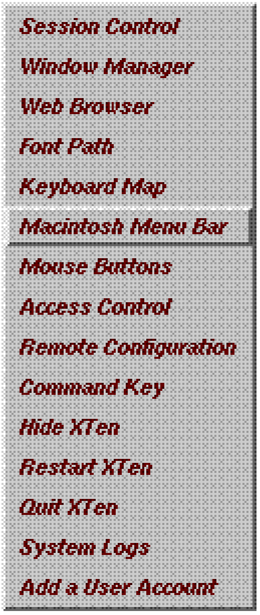
Select a Keyboard Mapping and click Save.

Keyboard Mapping

Your Keyboard Control Panel indicates the keyboard la intosh. If your keyboard layout is an international (non-U.S) layout, t out must be re-mapped when the X server is launched to match your country's l

Figure 18. Keyboard Mapping

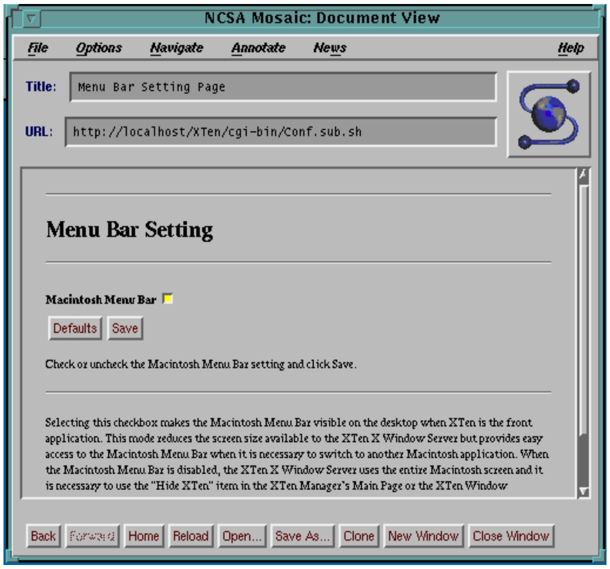
## 3.2.6 Setting the Macintosh Menu Bar



The Macintosh menu bar may be configured as visible or invisible when X$^{Ten}$ has control of the display. By default, the X$^{Ten}$ menu bar is configured as invisible to give the user a more realistic X Window experience. When visible, the X$^{Ten}$ menu bar appears above the X$^{Ten}$ desktop and controls X$^{Ten}$’s visibility on the desktop relative to other Macintosh applications that may be running. The X$^{Ten}$ menu bar provides the traditional Macintosh file menu with a “Quit” menu item. Selecting this item shuts down the X$^{Ten}$ application. Alternately, the OpenLook Virtual Window Manger’s Desktop Menu can be used to hide X$^{Ten}$ and, thus, make the Macintosh Finder desktop visible.

To make the Macintosh menu bar visible when X$^{Ten}$ is in the foreground, click on the “Macintosh Menu Bar” button. Then click the “Save” button to inform the X$^{Ten}$ Manger of your choice. If the Macintosh menu bar is hidden, you may quit X$^{Ten}$ using the <Command>-<Option>-Q key sequence.

The “XServer” and “XMachTen” HTML hot links are available for more information.



Title: Menu Bar Setting Page

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

Menu Bar Setting

**Macintosh Menu Bar** ☑

Defaults Save

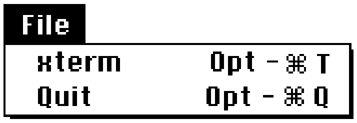
Check or uncheck the Macintosh Menu Bar setting and click Save.

Selecting this checkbox makes the Macintosh Menu Bar visible on the desktop when XTen is the front application. This mode reduces the screen size available to the XTen X Window Server but provides easy access to the Macintosh Menu Bar when it is necessary to switch to another Macintosh application. When the Macintosh Menu Bar is disabled, the XTen X Window Server uses the entire Macintosh screen and it is necessary to use the "Hide XTen" item in the XTen Manager's Main Page or the XTen Window

Figure 19. Menu Bar Setting

### 3.2.6.1 X$^{Ten}$ Menus

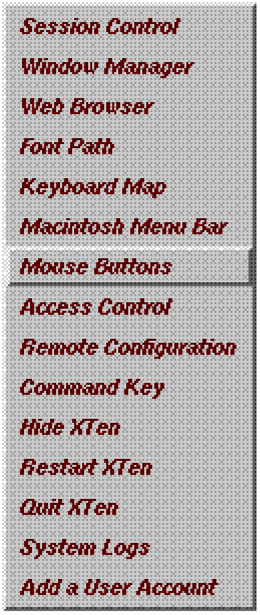
X$^{Ten}$ provides the traditional Macintosh File menu containing two items — "xterm" and "Quit". Each menu item has a hot key equivalent invoked by pressing <Command>-<Option>-T for "xterm", and <Command>-<Option>-Q for "Quit". As described below, each menu item is also affected by optionally pressing the <Shift> key in conjunction with selecting the menu item or the hot key equivalent. The hot key equivalents (with or without the <Shif>t key) will work whether or not the Macintosh menu bar is visible.



Invoking “xterm” will create a local instance of the X client *xterm*. However, if X$^{Ten}$ is being managed by a remote XDM session that does not permit local X clients to connect to the X server, xterm will not appear and no error message will be reported. Otherwise, xterm will appear and will be running locally on the X$^{Ten}$ system. If the <Shift> key is pressed in conjunction with the “xterm” menu item or hot key equivalent, the xterm that is started will also be the X$^{Ten}$ console device.

Invoking “Quit” will shut down X$^{Ten}$ in an orderly fashion by terminating each active X client, the X server, and the rest of the X$^{Ten}$ system. If the <Shift> key is pressed in conjunction with the “Quit” menu item or hot key equivalent, a fast shut down is done without waiting for each X client, the X server, or the rest of the X$^{Ten}$ system to complete their termination processing.

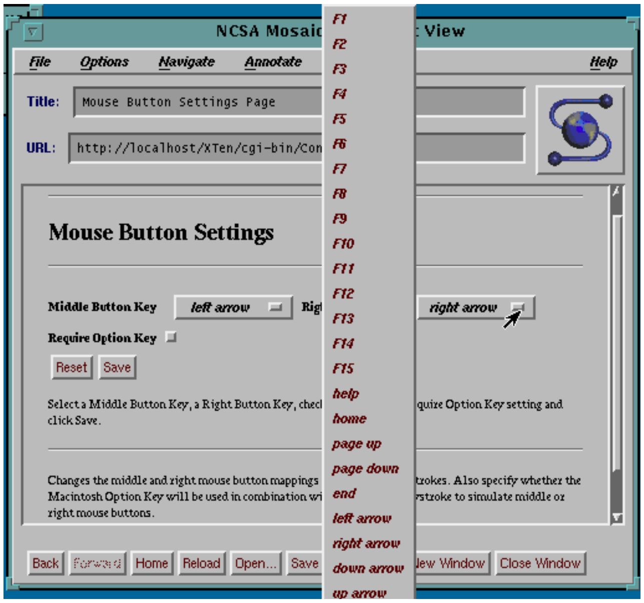
## 3.2.7 Mouse Button Mapping



Every Macintosh has a one-button mouse. The X11 standard prescribes a three-button mouse. Many window managers and X11 client applications depend on the ability to click with a second or third mouse button. You may purchase any commercial Macintosh-compatible three-button mouse to use with X$^{Ten}$. In the absence of a three-button mouse, the left arrow or right arrow keys are depressed to signify a middle or right mouse button click to X$^{Ten}$.

The Macintosh keyboard keys that are used to signify a second or third mouse button click can be changed to support personal preferences or various three-button mouse setting requirements. To reset the modifier for the middle button, click on the “Middle Button Key” menu and select an alternate key. To reset the key for the right button, click on the “Right Button Key” menu and select an alternative key.  If, in addition to the middle or right button key, you want to require the Macintosh <Option> key to be pressed simultaneously with the selected key, check the “Require Option Key” button. With these settings in place, click on the “Save” button to inform the X$^{Ten}$ Manager of your request. Note that X$^{Ten}$ will have to be restarted to incorporate these changes.

The “More Mouse Button Mapping Info” HTML hot link contains more detailed information on mouse button mapping.



Title: Mouse Button Settings Page

URL: http://localhost/XTen/cgi-bin/Con

Mouse Button Settings

Middle Button Key: left arrow
Rig: right arrow

Require Option Key □

Select a Middle Button Key, a Right Button Key, chec quire Option Key setting and click Save.

Changes the middle and right mouse button mappings trokes. Also specify whether the Macintosh Option Key will be used in combination wi ystroke to simulate middle or right mouse buttons.

Figure 20. Mouse Button Settings

## 3.2.7.1 Setting Up a Three-Button Mouse

Many X clients assume the mouse has three buttons. X^Ten simulates the middle and right mouse buttons with keystrokes. By default, the left arrow key generates middle button events and the right arrow key generates right button events. The real mouse button generates left button events.

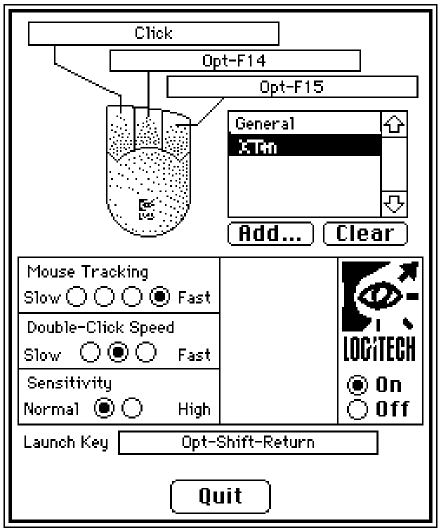
The original function of the arrow keys may be obtained by holding down the “Option” key while pressing one of the arrow keys. Normal arrow key functions are also restored when you re-map the middle and right mouse buttons using the “Mouse Buttons” configuration page.

Other references in this document to a mouse button refer to the Macintosh mouse button, which generates left button events.

***Three-Button Mouse Configuration***. Several commercial three-button mice are offered for the Macintosh. All operate by assigning an unused keyboard key or key combination to the middle and right mouse buttons. X$^{Ten}$ supports the reconfiguration of the mouse keys from the default left arrow and right arrow keys to any other keyboard key or <Option-*key*> combination. By matching the X$^{Ten}$ mouse button configuration with your three-button mouse hardware configuration, you can shift your X three-button operation entirely to the hardware mouse.

If the default mouse button configuration is modified or you use the Macintosh <Option> key in combination with another key to simulate a mouse button, all arrow keys revert to their original functions.

***Example Three-Button Mouse Configuration***. Suppose you own a Logitech Mouse and would like to map the middle and right mouse buttons to the <Option-F14> and <Option-F15> key combinations when running X$^{Ten}$. Your Logitech MouseMan™ control panel would look something like this:



Click

Opt-F14

Opt-F15

General

XTen

Mouse Tracking

Slow ○ ○ ○ ● Fast

Double-Click Speed

Slow ○ ● ○ Fast

Sensitivity

Normal ● ○ High

● On

○ Off

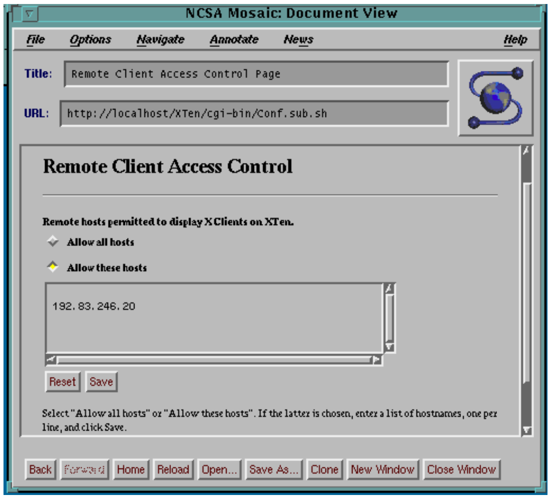
Launch Key Opt-Shift-Return

**Figure 21. Logitech MouseMan™ Control Panel**

## 3.2.8 Access Control Permissions

It is important to pay attention to data security in a distributed network environment. This configuration item allows you to explicitly list the host names or host addresses that may display X clients on the local X$^{Ten}$ display. Alternatively, you may specify that any host may display X client requests on the local display.

Add or delete the host name or host address entries[1] that you wish to allow or disallow access to the local display. If you wish to allow any host to access the local display, click the “Allow All Hosts” button. When the modifications are complete, click the “Save” button to inform the X$^{Ten}$ Manager of your selection. You must restart X$^{Ten}$ for these changes to take effect.



Title: Remote Client Access Control Page

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

Remote Client Access Control

Remote hosts permitted to display X Clients on XTen.

○ Allow all hosts

● Allow these hosts

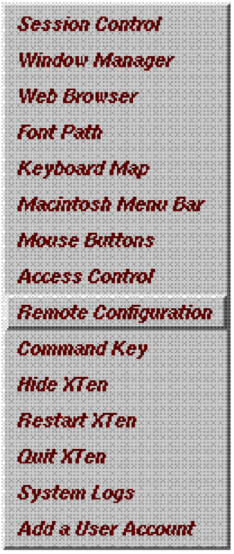
```
192.83.246.20
```

Select "Allow all hosts" or "Allow these hosts". If the latter is chosen, enter a list of hostnames, one per line, and click Save.

**Figure 22. Remote Client Access Control**

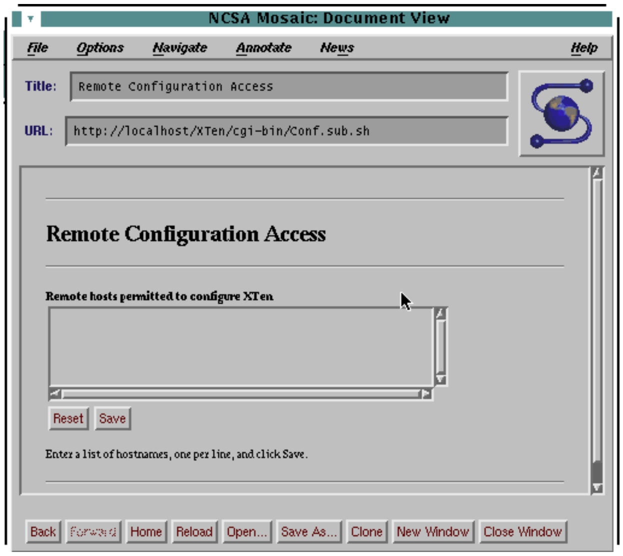
[1] Using internet “dot” addresses instead of host names avoids dependence on name server resolution on X$^{Ten}$ startup.

## 3.2.9 Remote Configuration Permissions



The "Remote Configuration" item limits the hosts that are permitted to remotely configure X$^{Ten}$ by communicating with the X$^{Ten}$ Manager. ("Access Control" controls the external hosts that are allowed to display X clients on the local system.)

Add or delete the host names or host addresses to control their ability to remotely configure the local X$^{Ten}$. Click on the "Save" button to inform the X$^{Ten}$ Manager of your choice. You must restart X$^{Ten}$ for these changes to take effect.



Title: Remote Configuration Access

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

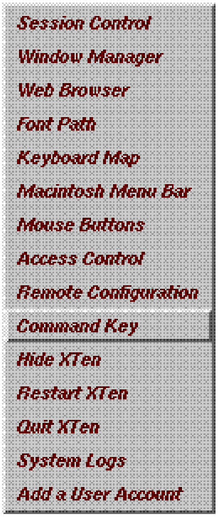
Remote Configuration Access

Remote hosts permitted to configure XTen

Enter a list of hostnames, one per line, and click Save.

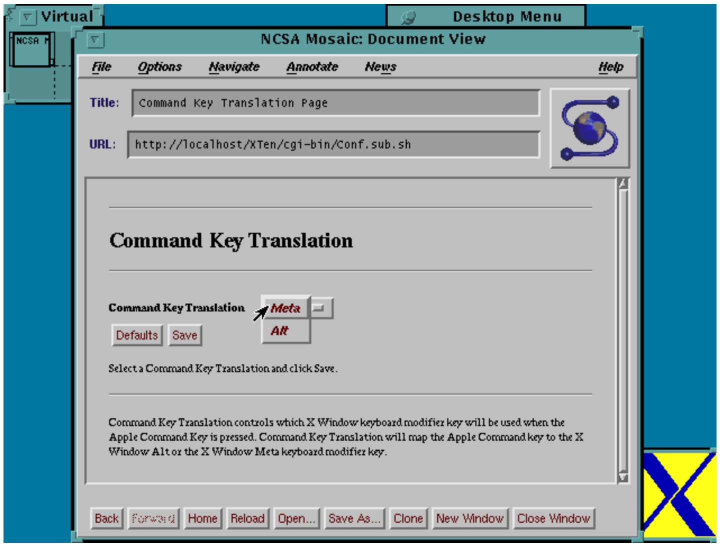
Figure 23. Remote Configuration Access

## 3.2.10 Command Key



***X Meta and Alt Keys***. Under X$^{Ten}$, the Macintosh <Command> key functions by default as an "X Meta" (or when shifted, "Alt") key. Many applications, including the Motif Window Manager, use the Meta and Alt keys as keyboard modifiers. Use the configuration manager to change the default operation of the <Command> key to Alt (or when shifted, Meta).

Apple reserves certain keystroke combinations for special effects. All <Command>-<Shift>-<*number*> sequences are reserved for use by MacOS and, therefore, are not available for use by X$^{Ten}$ or X applications.



Title: Command Key Translation Page

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

Command Key Translation

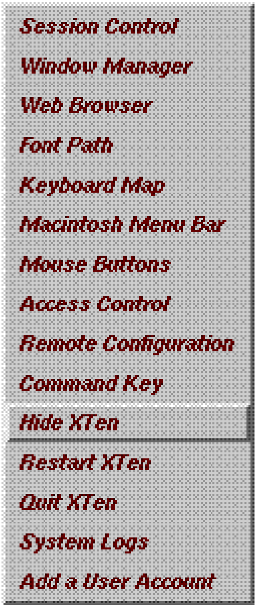
Command Key Translation: Meta

Select a Command Key Translation and click Save.

Command Key Translation controls which X Window keyboard modifier key will be used when the Apple Command Key is pressed. Command Key Translation will map the Apple Command key to the X Window Alt or the X Window Meta keyboard modifier key.

Figure 24. Command Key Translation

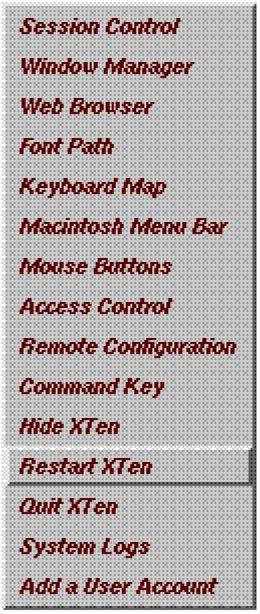
### 3.2.11 Hiding X$^{Ten}$



The “Hide XTen” menu item hides the X$^{Ten}$ desktop and reverts control of the display, mouse and keyboard to the Macintosh Finder. The traditional Macintosh Finder desktop is displayed and full Finder operations are restored.

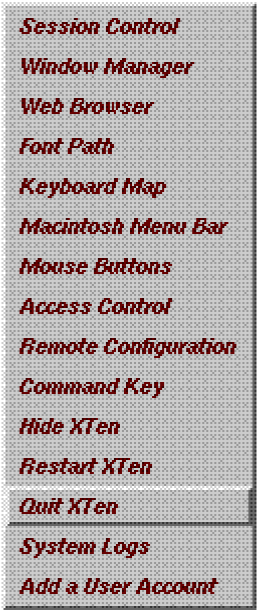
Access to the X$^{Ten}$ desktop can be regained by selecting the X$^{Ten}$ application from the Finder application menu in the upper right-hand corner of the Finder desktop. When X$^{Ten}$ is selected, the Finder desktop will be hidden and the X$^{Ten}$ desktop will once again take prominence.

## 3.2.12 Restarting X$^{Ten}$



The “Restart XTen” menu item will restart the X$^{Ten}$ application. If you make a change in configuration, the X$^{Ten}$ application must be restarted to incorporate the new configuration into its operation. This menu item provides for that function automatically.

## 3.2.13 Quitting X^Ten

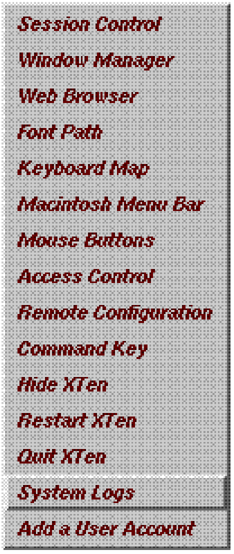


The “Quit XTen” menu item will quit the X^Ten application and return the user to the Macintosh Finder desktop.

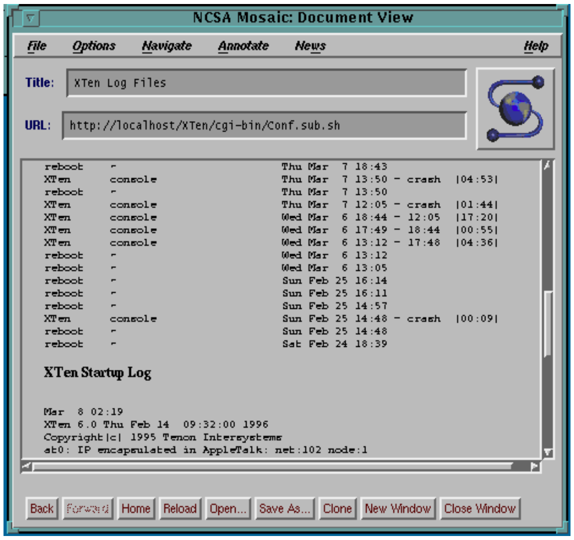
If X^Ten is using the Tenon TCP stack (see “Appendix G — Using OpenTransport or Replacing OpenTransport”), all MacTCP applications before quitting X^Ten. Before launching subsequent MacTCP applications, you must either reboot your system or restart X^Ten.

If X^Ten is using OpenTransport, there are no dependencies on the order of quitting and restarting other MacTCP applications.

## 3.2.14 System Logs



A number of elements within the X$^{Ten}$ multi-threaded execution environment make status or error information available in different log files. This X$^{Ten}$ Manager page provides access to most of those log files. Files that are logged include X Server Start up Errors, X Server Errors, Launch Errors, XTen Usage Log, XTen Startup Log and the XTen Startup Configuration Log. These log files are extremely helpful in the event of X$^{Ten}$ system or X client failure.



Title: XTen Log Files

URL: http://localhost/XTen/cgi-bin/Conf.sub.sh

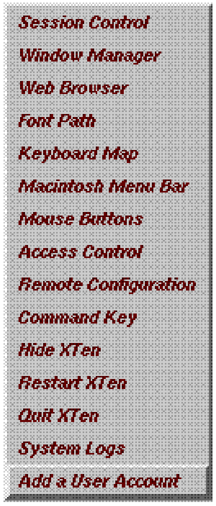
```
reboot     ~                          Thu Mar  7 18:43
XTen       console                    Thu Mar  7 13:50 - crash  |04:53|
reboot     ~                          Thu Mar  7 13:50
XTen       console                    Thu Mar  7 12:05 - crash  |01:44|
XTen       console                    Wed Mar  6 18:44 - 12:05  |17:20|
XTen       console                    Wed Mar  6 17:49 - 18:44  |00:55|
XTen       console                    Wed Mar  6 13:12 - 17:48  |04:36|
reboot     ~                          Wed Mar  6 13:12
reboot     ~                          Wed Mar  6 13:05
reboot     ~                          Sun Feb 25 16:14
reboot     ~                          Sun Feb 25 16:11
reboot     ~                          Sun Feb 25 14:57
XTen       console                    Sun Feb 25 14:48 - crash  |00:09|
reboot     ~                          Sun Feb 25 14:48
reboot     ~                          Sat Feb 24 18:39
```

XTen Startup Log

```
Mar  8 02:19
XTen 6.0 Thu Feb 14  09:32:00 1996
Copyright|c| 1995 Tenon Intersystems
at0: IP encapsulated in AppleTalk: net:102 node:1
```

Figure 25. System Logs

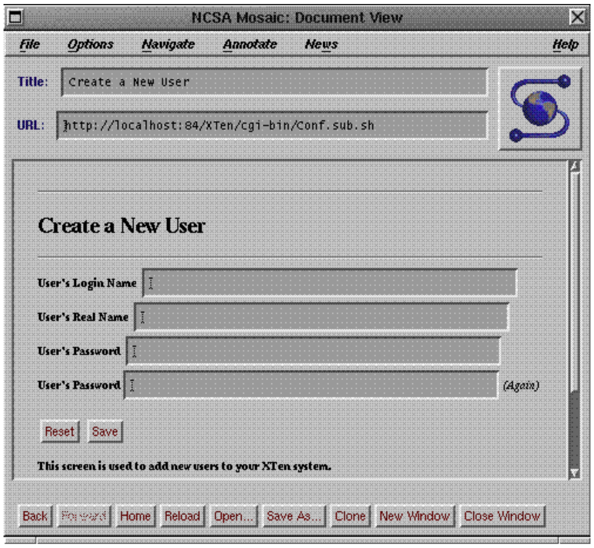
## 3.2.15 Adding User Accounts



X$^{Ten}$ can support multiple user accounts. These accounts may be used by different people who share time on the same X$^{Ten}$ system; for example, in a student computer lab or by different people who log on to the X$^{Ten}$ system from another computer or X terminal.

To add a user account to X$^{Ten}$, simply enter the account name (or login ID), the user's real name, and the password. The password is not echoed back in the browser. It must be entered twice to catch typing errors. When all of the fields are filled in, click "Save".

By default, X$^{Ten}$ does not require a user name and password. For information about configuring X$^{Ten}$ to require a valid user name and password, see section"6.2.4 X Display Management Under X$^{Ten}$".



Title: Create a New User

URL: http://localhost:84/XTen/cgi-bin/Conf.sub.sh

Create a New User

User's Login Name

User's Real Name

User's Password

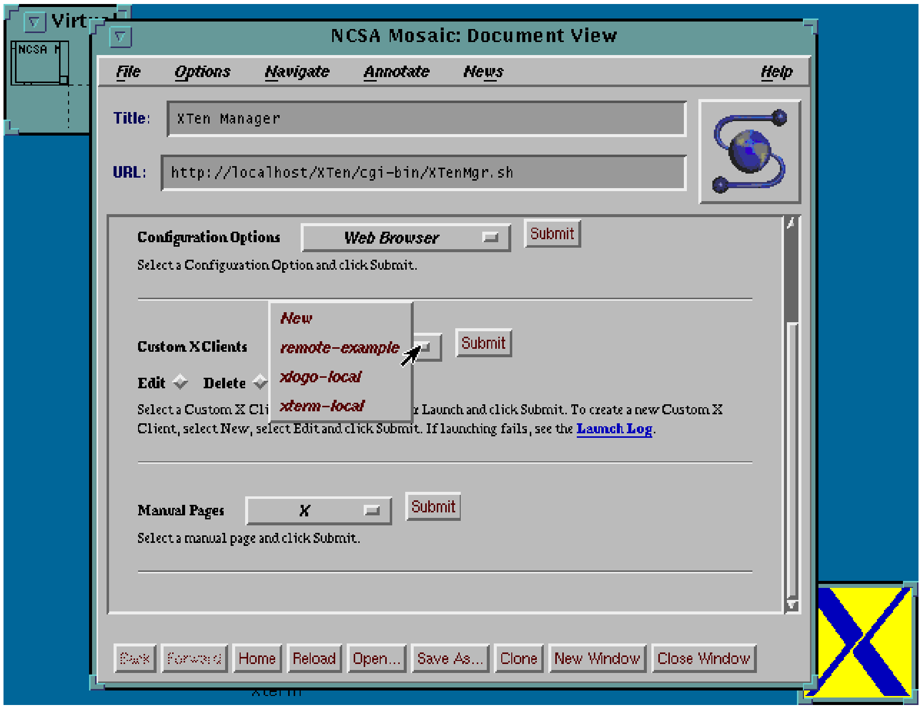
User's Password (Again)

This screen is used to add new users to your XTen system.

Figure 26. Create a New User

## 3.3 Customizing X Clients

The third X$^{Ten}$ Manager component provides support for configuring local and remote X clients. Once an X11 client is configured, the name given to the configuration appears in the “Custom X Clients” sub-menu of the OpenLook Desktop Menu. Selecting a client from that sub-menu will cause the custom client defined here to be executed. Each of the defined custom clients will also appear as a selectable menu item in the X$^{Ten}$ Manager “Custom X Clients” menu. Using the “Custom X Clients” menu, you can edit, delete or launch existing clients or, by choosing “New”, create a new client. Once you have made your selection, click “Submit” to activate that selection.



Title: XTen Manager

URL: http://localhost/XTen/cgi-bin/XTenMgr.sh

Configuration Options: Web Browser

Select a Configuration Option and click Submit.

Custom X Clients

Edit ◇ Delete ◇

Select a Custom X Cli r Launch and click Submit. To create a new Custom X Client, select New, select Edit and click Submit. If launching fails, see the Launch Log.

Manual Pages: X

Select a manual page and click Submit.

Figure 27. Custom X Clients

### 3.3.1 Client Editor

The "Client Editor" is used to create new and edit existing X client startup scripts. Either local or remote X clients may be specified, along with parameters for their execution.

Each client is given an arbitrary name under the "Client Name" entry. This name will appear in the "Custom X Clients" menu in the OpenLook Desktop Menu. Selecting a name from the "Custom Clients" list will cause the contents of the script input into the "Command" entry to be executed either locally or on a designated remote host.

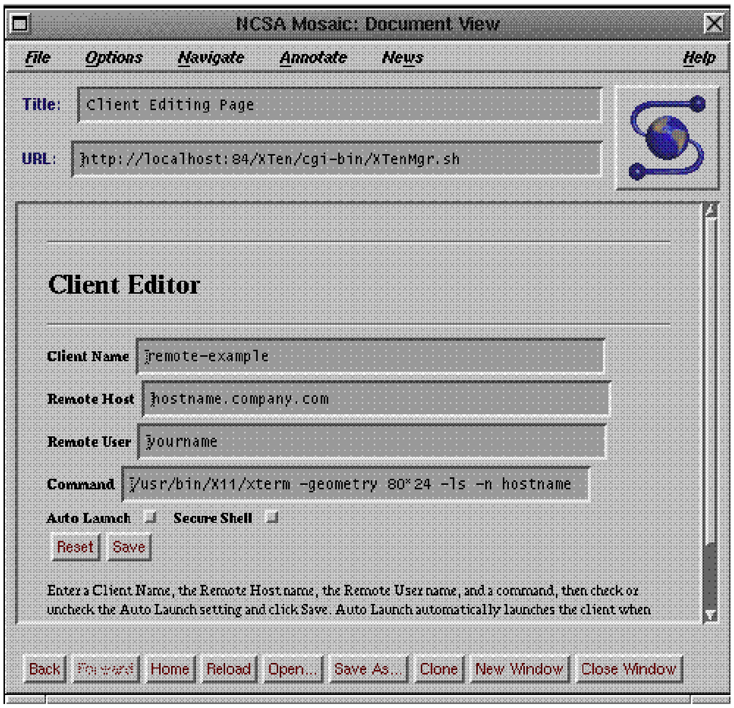
If the "Remote Host" and "Remote User" entries are left blank, the client is assumed to be a locally executed X client. If the "Remote Host" and "Remote User" entries are filled in, the remote host is contacted using the UNIX *rsh* application. The "Remote User" name is attached to the *rsh* request, along with the "Command" script information. The local user name attached to the *rsh* request is "XTen", or use the name of the logged in user.

The results of the execution are in the Launch Log and may be viewed by clicking the "Launch Log" HTML hot link.

After a client is specified, select the "Auto Launch" button if you want the client to be automatically executed whenever X$^{Ten}$ is started. Finally, click on the "Save" button to inform the X$^{Ten}$ Manager of your configuration change.

X$^{Ten}$ also supports the execution of remote X clients via an encrypted stream using Secure Shell (*ssh*). To specify an *ssh* connection, click on the "Secure Shell" button before clicking "Save".

Before you can launch remote X clients using *rsh* or *ssh*, you may be required to set up and configure the Remote Shell daemon (*rshd*) or the Secure Shell daemon (*sshd*) on the remote system. Refer to section "5.0 Executing Remote X Clients" for more information on using *rsh* and *ssh* to launch remote clients.



Title: Client Editing Page

URL: http://localhost:84/XTen/cgi-bin/XTenMgr.sh

## Client Editor

Client Name: remote-example

Remote Host: hostname.company.com

Remote User: yourname

Command: /usr/bin/X11/xterm -geometry 80x24 -ls -n hostname

Auto Launch □ Secure Shell □

Reset Save

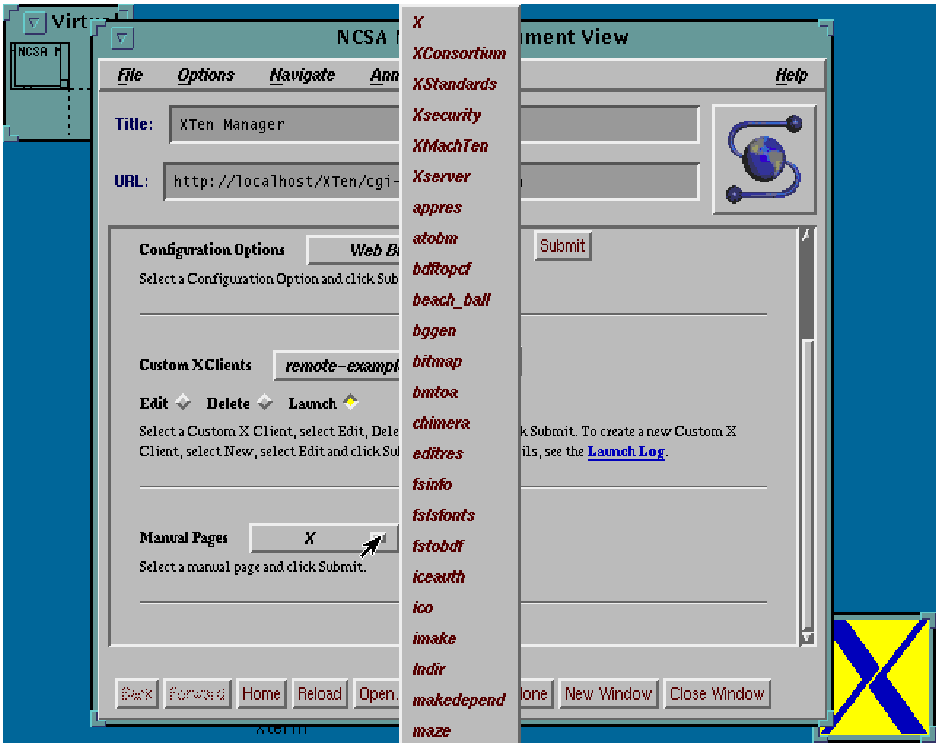
Enter a Client Name, the Remote Host name, the Remote User name, and a command, then check or uncheck the Auto Launch setting and click Save. Auto Launch automatically launches the client when

Figure 28. Client Editor

## 3.4 Viewing X$^{Ten}$ "Manual" Pages

The X$^{Ten}$ Manager has a fourth component that contains an extensive selection of online electronic reference information. Only manual pages that have been installed will appear. To access this information, select a manual page and click "Submit" to display the information.

In addition, an *xman* X client is included with X$^{Ten}$.



Title: XTen Manager

URL: http://localhost/XTen/cgi-

Configuration Options

Select a Configuration Option and click Sub

Custom X Clients remote-exampl

Select a Custom X Client, select Edit, Dele k Submit. To create a new Custom X Client, select New, select Edit and click Sul ils, see the Launch Log.

Manual Pages X

Select a manual page and click Submit.

Figure 29. Manual Pages

# 4.0 The X Desktop

X$^{Ten}$ is a multi-threaded Macintosh application that creates an alternative desktop (an “X desktop”) on your display. With X$^{Ten}$ you will have two desktops — the X desktop and your Macintosh Finder desktop. X application output will appear on the X desktop, while Macintosh application windows will appear on the Macintosh desktop. You will be able to easily toggle between these two desktops and use any Macintosh or X applications.

The X$^{Ten}$ X client applications and the X window manager define the look and feel of an X desktop. The window manager is a program that runs within X$^{Ten}$ and manages the windows on your screen. A window manager provides functions for opening, closing, moving and resizing windows. Window managers display a three-dimensional frame around each window. Without a window manager, your windows would not have borders and it would be difficult, if not impossible, to perform many simple window functions.

X$^{Ten}$ provides you with a choice of the AfterStep Window Manager (*afterstep*), the OpenLook Virtual Window Manager (*olvwm*), the Motif Window Manager (*mwm*) and the Tab Window Manager (*twm*). By default, the AfterStep Window Manager is launched when X$^{Ten}$ is started.

# 4.1 Navigating the X Desktop

The AfterStep, OpenLook and Motif Window Managers contain sophisticated point-and-click desktop operations that are important to explore. This section contains a brief discussion of basic mouse and window operations applicable to any window manager, and then shows you how to navigate the AfterStep, OpenLook and Motif desktops.

## 4.1.1 Basic Desktop Operations

This section contains information on the basic manipulation of $X^{Ten}$ windows. These functions include mouse usage, window selection, window movement, window resizing, window iconification and uncovering or raising a window.

### 4.1.1.1 Using the Mouse

As you move the mouse on your X desktop, the pointer on the screen (the cursor) moves correspondingly.

Whenever this section tells you to "point to" something, it simply means move the mouse until the pointer on your screen is positioned over the "something".

***Pointer Shapes***. The location of the pointer can cause the shape of the pointer to change. For example, when the pointer is directly over the root window (the backdrop behind all windows), the pointer has an arrow shape. When you point to the inside of a terminal window, the pointer changes to an "I" shape. Different pointer shapes are shown in "Figure 30. Pointer Shapes".

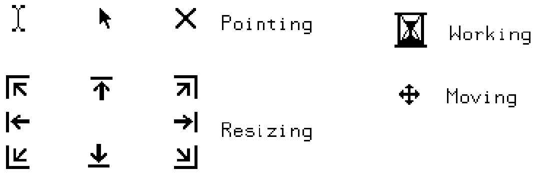


**Figure 30. Pointer Shapes**

***Clicking, Double-Clicking and Dragging***. The following terminology is used to refer to actions involving mouse buttons:

- "click" means to press and release a button without moving the pointer
- "double-click" means to click a button twice in rapid succession
- "drag" means to press and hold a button while moving the pointer

## 4.1.1.2 Selecting a Window

Before a window can receive keyboard input, it must be selected as the active window. To select a window, point to any part of the window and click the select button on the mouse. The select button is your mouse button, or the left button on a three-button mouse.

A window frame changes appearance when the window is active. When you press on the keyboard, the characters appear in the active window.

Keyboard input is directed to the window containing the cursor. If no window is active or if the cursor is on the X desktop, everything you type is lost. If the cursor is not in the window to which you want input directed, input will be lost or transmitted to another window containing the cursor.

### 4.1.1.3 Quitting the Window Environment

Before exiting the window environment, exit any application programs and stop any commands that may be running in terminal windows.

Pull down the File menu item from the X$^{Ten}$ menu bar and select "Quit". If the menu bar is invisible, press the <Command>-<Option>-Q keys simultaneously to quit X$^{Ten}$. This removes all terminal windows from the screen (as well as the windows for clocks and similar windowed programs) and terminates X$^{Ten}$. (See also section "3.2.13 Quitting X$^{Ten}$".)

### 4.1.1.4 Moving a Window

Using the pointer, you can move a window to a new location on the screen. To do so, grab the window's title bar with the pointer, drag it to a new location, and then release the window.

*Step 1 — Grab the Title Bar with the Pointer*

Point to the title bar of the window. The title bar is the rectangular area across the top of the window where the name of that window appears. When properly positioned, the pointer is an arrowhead as shown in "Figure 31. Grabbing the Title Bar".

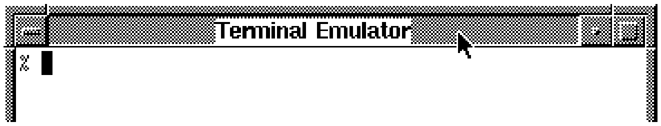


**Figure 31. Grabbing the Title Bar**

When the pointer is positioned, press and hold the select button.

### *Step 2 — Drag the Window to a New Location*

While holding the select button, slide the mouse to the right across your desktop. Now slide the mouse toward you.

As you move the mouse, the pointer on the screen drags an outline of the terminal window as shown in "Figure 32. Repositioning the Window". The outline shows you where the window will be moved when you release the mouse button.

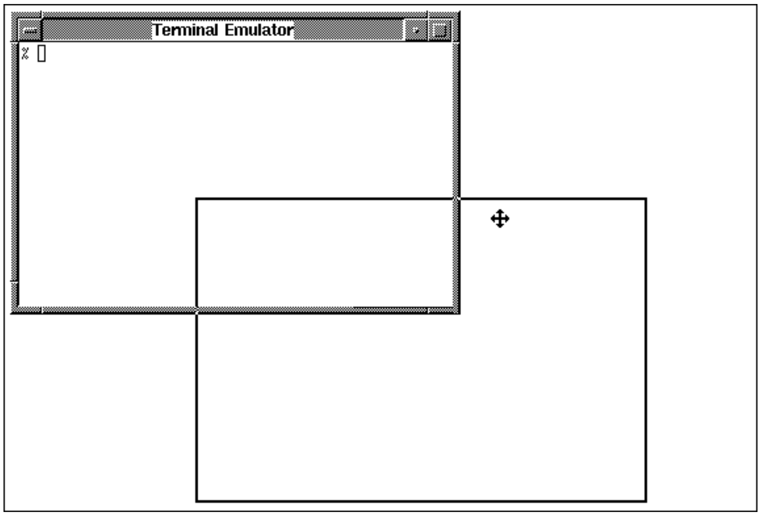


**Figure 32. Repositioning the Window**

### *Step 3 — Release the Window*

Move the outline to the center of the screen. When the outline is where you want it for now, release the mouse button. The window moves to the new position.

## 4.1.1.5 Changing the Size of a Window

You can change the size of a window by grabbing an area of the window’s frame with the pointer, dragging the frame to the desired size, and then releasing the frame.

### *Step 1 — Grab the Frame with the Pointer*

Where you grab the window’s frame determines how the window will be resized.

In AfterStep, point and click on either bottom corner of a window or the middle of the bottom of a window and the window can be stretched or shrunk vertically and horizontally.

In OpenLook, point and click on any corner of a window and the window can be stretched vertically and horizontally at the same time. The window is anchored at the corner diagonal to the one grabbed by the mouse. A window can be grown or shrunk as long as the mouse key is held down. When the key is released, the window resizing is complete. In order to type input to the window, the mouse must be positioned within the newly resized window area.

Window sizing is slightly different with Motif. Windows can be sized by grabbing an edge, as well as by grabbing a corner. The following table enumerates the different resizing tactics that are available with a Motif window.

To resize a window, point to the lower right corner of the window frame. When positioned correctly, the pointer’s shape changes to a circle in OpenLook, and to an arrowhead pointing into the corner of the frame in Motif. Grab the frame by pressing and holding down the select button.

### Places to Grab a Motif Window Frame

| If you want to stretch or shrink the window ... | point to ... |
|---|---|
| vertically from the ... | |
| top | top of the frame, above the title bar |
| bottom | bottom of the frame |
| horizontally from the ... | |
| right | right side of the frame |
| left | left side of the frame |
| diagonally from the ... | |
| bottom left corner | frame’s lower left corner |
| top left | frame’s upper left corner |
| top right | frame’s upper right corner |
| bottom right | frame’s lower right corner |

### *Step 2 — Enlarge or Shrink the Window*

With the select button still pressed, slide the mouse so that the pointer moves away from the window. Now slide the mouse so the pointer moves toward the window. As you move the mouse, the pointer on the screen drags an elastic outline of the window. The outline shows you the new size of the window.

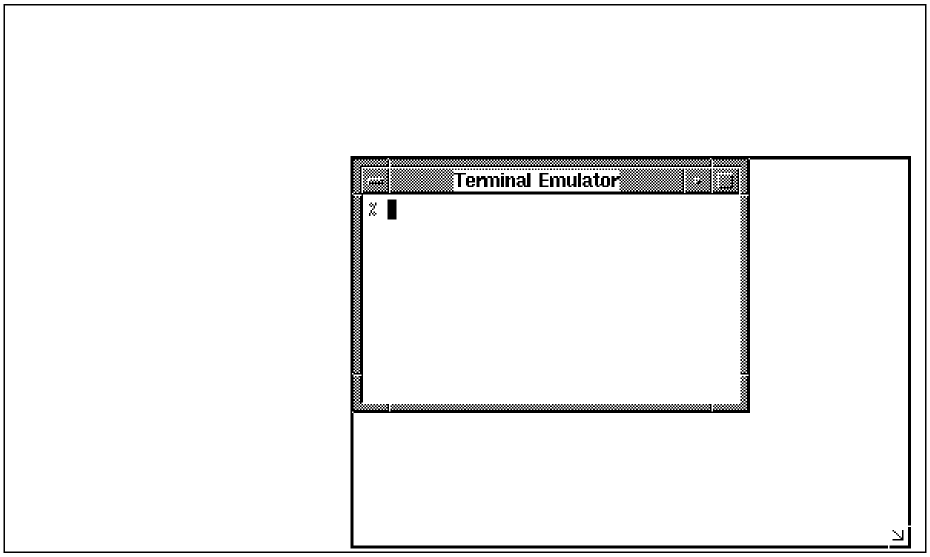


**Figure 33. A Window and Its Elastic Outline**

Now, stretch the window until it reaches the lower right corner of the root window.

### *Step 3 — Release the Frame*

When the elastic outline is the correct size, release the mouse button. The screen is redrawn with the window filling the outline.

### 4.1.1.6 Changing a Window Into an Icon

As you work, your screen can become cluttered with windows. Changing a few of those windows into icons allows you to tidy up a cluttered work space. Icons are small graphical images. Programs executing in a minimized window continue to execute until they finish or halt because they require input from you.

#### *Step 1 — Locate the Minimize Button*

Slide the mouse so that you position the pointer on the minimize button. In AfterStep and OpenLook the minimize button is a square on the left end of the title bar; in Motif, the minimize button is the small square to the immediate right of the title bar on the window frame as shown below.

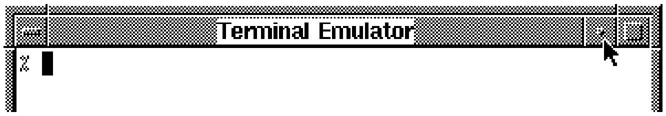
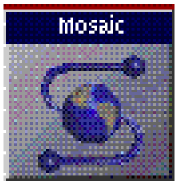


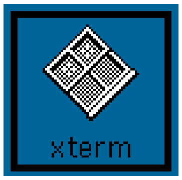
Figure 34. Motif Title Bar Minimize Button

#### *Step 2 — Press the Minimize Button*

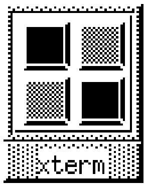
Press the minimize button by clicking the select button. The window changes into an icon. Icons are displayed in a row along the bottom of the root window, starting on the left as illustrated below. Example AfterStep, OpenLook and Motif icons are shown below.



AfterStep Icon



OpenLook Icon



Motif Icon

## 4.1.1.7 Moving an Icon

To move an icon around the screen, follow these steps:

1. Point to the icon.

2. Press and hold the select button.

3. While holding the select button, drag the pointer. An outline follows the pointer, showing where the icon will be repositioned.

4. To position the icon, release the mouse button.

In some window managers, icons are positioned on a grid in some window managers, so the icon you move may jump a bit if you set it too close to the margin of the root window or too close to another icon.

## 4.1.1.8 Restoring a Window From an Icon

To restore an icon (change it back into a window), point to the icon and double-click the select button. To double-click a mouse button, press the button twice in rapid succession.

### *Step 1 — Point to an Icon to Restore*

Point to the icon by positioning the mouse pointer on the icon. If you are unable to identify the icon you want to restore because the icon's name is obscured within the icons' title area, you can display an entire label by clicking the select button in an icon's title.

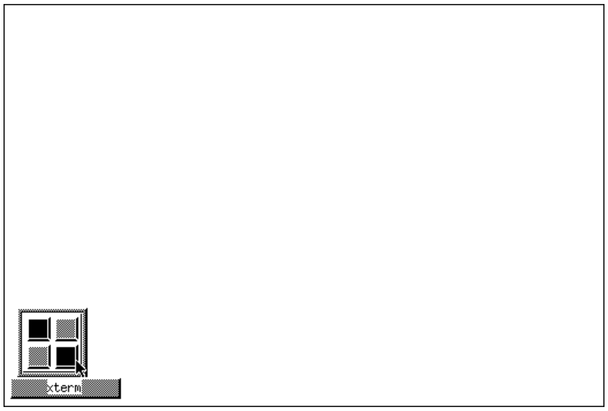
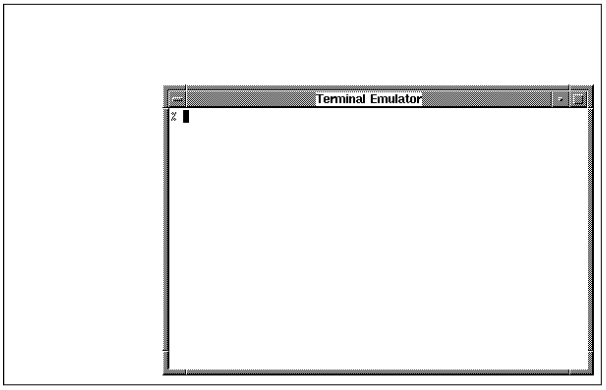


**Figure 35. Motif Icon and Its Pop-Up Label**

### *Step 2 — Change the Icon Into a Window*

While pointing to the icon, double-click the select button. The icon changes back into the window from which it originated. The window positions itself where it was before being minimized, as shown in “Figure 36. Restored Window”.



```
%
```

**Figure 36. Restored Window**

### 4.1.1.9 Raising a Window

To bring a partially concealed window to the front of the root window, click on the window's frame.

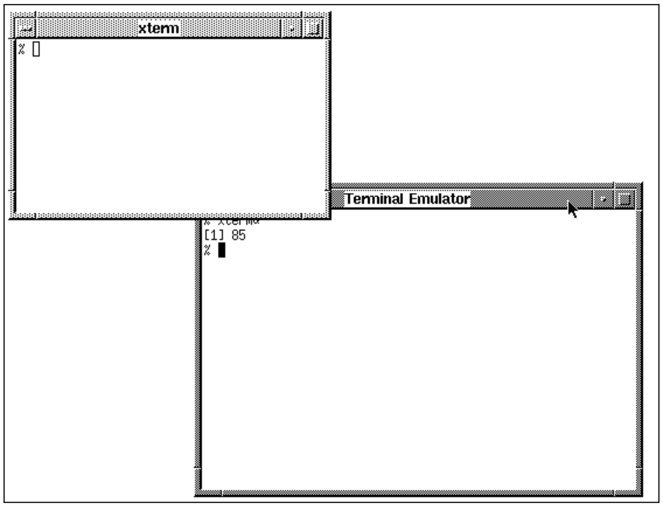
To demonstrate this, create another terminal window by typing the following command in the existing terminal window:

```
    xterm & <CR>
```

The new window appears in the upper left corner of the display, partially covering the first window (as shown in the "Figure 37. Partially Concealed Window").

#### *Step 1 — Position the Pointer*

Position the pointer on any visible portion of the concealed window's frame.



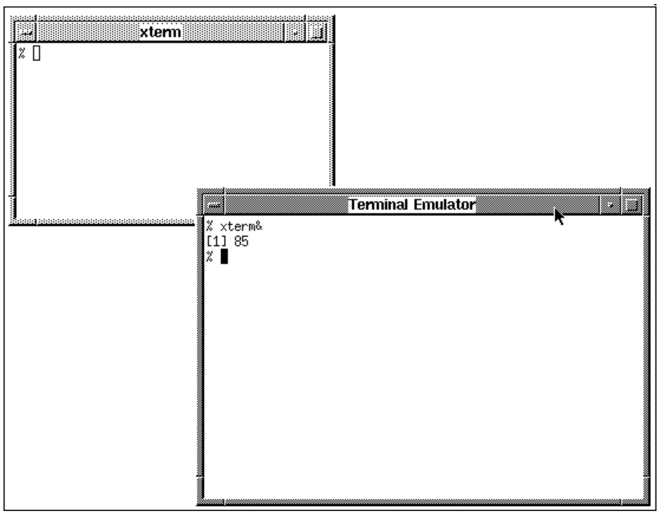
```
% ▯
```

```
% xterm&
[1] 85
% ▮
```

Figure 37. Partially Concealed Window

### *Step 2 — Press the Select Button*

Click the select button on the mouse. The partially concealed window moves to the front of all other windows, as shown in "Figure 38. Window Revealed".



```
%
```

```
% xterm&
[1] 85
%
```

**Figure 38. Window Revealed**

## 4.2 Navigating the AfterStep Desktop

The AfterStep Window Manager (*afterstep*) is the default window manager for local X sessions in $X^{Ten}$. The choice of a window manager is often based on personal preference and past experience. $X^{Ten}$ provides several alternate window managers including the OpenLook Window Manager (*olvwm*) and the Motif Window Manager (*mwm*). The $X^{Ten}$ Manager provides an easy-to-use mechanism to switch window managers. See section "3.2.2 Choosing a Window Manager" for details. If you use $X^{Ten}$ to connect to a remote X system (using XDM), the default window manager will depend on the settings on that system, and the selection of alternate window managers may be different than the set provided by $X^{Ten}$.

Each of $X^{Ten}$'s window managers has $X^{Ten}$-specific customizations in the default menu settings. Most of these customizations are similar between the different window managers, but the OpenLook Window Manager's desktop has had the most customizations for $X^{Ten}$. Since the OpenLook Window Manager contains a superset of the $X^{Ten}$ customizations, the menu items that are common between $X^{Ten}$'s window managers are documented under section "4.3 Navigating the OpenLook Desktop".

### 4.2.1 AfterStep Wharf

The AfterStep desktop includes a program wharf along the right-hand side of the screen. The wharf provides a row of icons that can be clicked to launch specific X clients. The contents of the wharf are controlled by AfterStep's user configuration file (*~.steprc)* and system configuration file (*/usr/lib/X11/afterstep/system.steprc*), and can be customized on a per-user or system level.

X$^{Ten}$'s initial wharf settings include a mechanism to lock the X desktop, a clock and calendaring program, a program monitoring the load on the X$^{Ten}$ system, X terminal windows, a Web browser, a virtual desktop manager and the AfterStep configuration manager.

To activate a program from within the wharf, simply click on its icon.

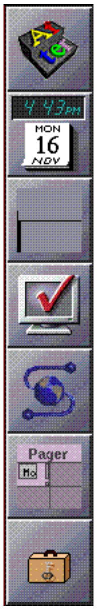


**Figure 39. AfterStep Wharf**

## 4.2.2 AfterStep Desktop Menu

The AfterStep Desktop Menu is accessible by clicking on the left mouse button anywhere in an empty portion of the AfterStep desktop. This menu provides a pull-down list of X clients that can be launched simply by selecting that item in the pull-down menu. (To select an item in a menu, stop the cursor on a menu item and release the mouse button.)

The initial X$^{\text{Ten}}$ settings include a mechanism to launch the X$^{\text{Ten}}$ Manager, a quick way to hide X$^{\text{Ten}}$ and return to the Macintosh Finder, X terminal windows, a list of other applications and utilities, a list of background screens, menus for the AfterStep wharf, pager, banner and clock modules, and some X$^{\text{Ten}}$ session control items like restarting AfterStep, quitting the X session and quitting X$^{\text{Ten}}$.

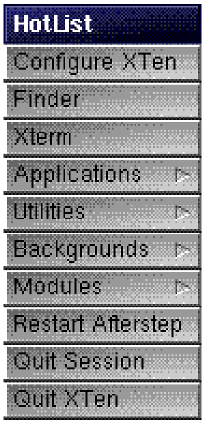


**Figure 40. AfterStep Desktop Menu**

### 4.2.3 AfterStep Current Applications Menu

The AfterStep Current Applications menu is accessible by clicking on the middle mouse button anywhere in an empty portion of the AfterStep desktop. This menu provides a pull-down list of X clients that are currently running. Note that the list includes clients that are displayed in other virtual desktops. Selecting a client from this menu brings that client to the foreground; i.e., it makes it the front-most client on the display. If the client was displayed in a different virtual desktop, that desktop is also displayed. This menu provides a quick way to locate and display any X client currently running on your system.

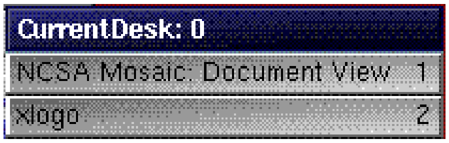


Figure 41. AfterStep Current Applications Menu

### 4.2.4 AfterStep Window Menu

The AfterStep Window Ops menu is accessible by clicking on the middle mouse button in the title bar of any AfterStep window or by clicking the right mouse button anywhere in an empty portion of the AfterStep desktop. If the latter method is used, the Window Ops menu applies to the next window that is clicked on after the window operation is selected.

The Window Ops menu provides a pull-down list of window operations that can be performed simply by selecting that item from the pull-down menu. The list includes operations for moving and resizing windows, as well as ordering, iconifying and closing windows.

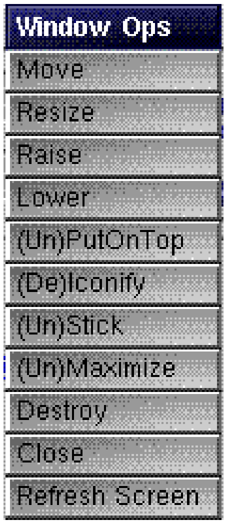


Figure 42. AfterStep Window Ops Menu

## 4.3 Navigating the OpenLook Desktop

Using the third mouse button, display the OpenLook Desktop Menu. (In the default configuration, the menu is "pinned" to the desktop.) With the mouse button depressed, scroll down each of the menu selections until the "Xterm" selection is highlighted, release the mouse button, and start an xterm client. For information on the third mouse button, see section "3.2.7 Mouse Button Mapping". By default, the third mouse button is mapped to the right arrow key.

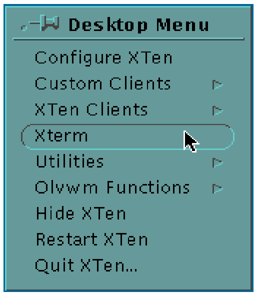
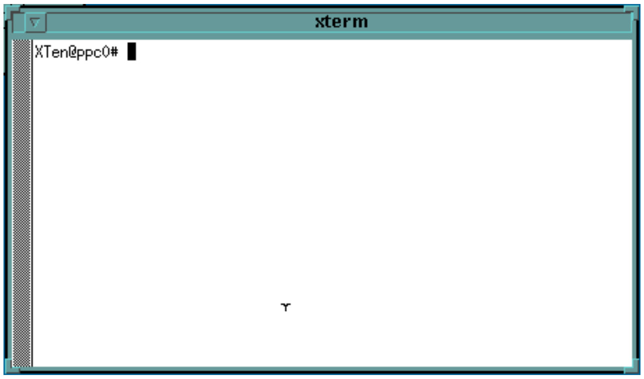


Figure 43. OpenLook Desktop Menu

This will create an *olvwm* xterm client. The OpenLook Window Manager (*olvwm*) xterm is shown below.



```
XTen@ppc0#
```

Figure 44. OpenLook xterm Window

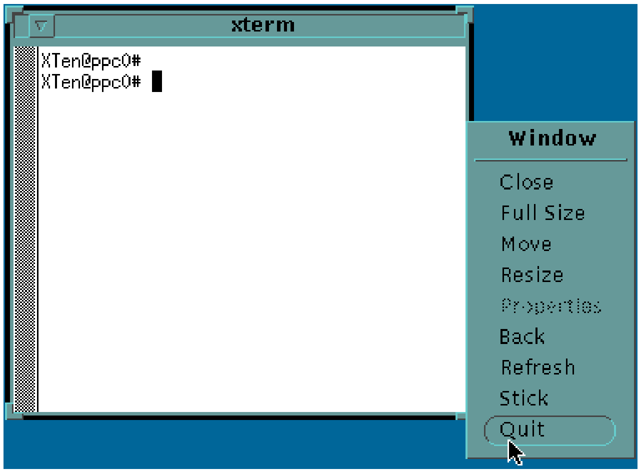
### 4.3.1 olvwm Window Menu

Each *olvwm* window has a Window menu that contains functions for controlling the window and, in the case of the "Quit" menu selection, its underlying X client.

#### *Step 1 — Display the Window Menu*

The *olvwm* Window menu is attached to the window frame.

In *olvwm*, position the mouse on an edge of a window and depress the third mouse button. With the third mouse button depressed, a small window will appear offering *olvwm* functions which may be applied to the window and its associated X client.



```
XTen@ppc0#
XTen@ppc0#
```

**Figure 45. olvwm Window Menu**

### *Step 2 — Choose a Function from the Menu*

While continuing to hold down the mouse select button, drag the pointer down the menu. As the pointer moves, it highlights a button for each available selection. Drag the pointer until you highlight the "Full Size" function. Release the mouse button.

The "Full Size" function causes the window to expand to fill the entire screen.

### *Step 3 — Restore the Window to its Original Size*

Display the *olvwm* Window menu again. Drag the pointer down the menu until you highlight the "Resize" selection. Release the mouse button. The window is highlighted. While holding down the mouse button, decrease the size of the window by dragging a variable outline of the window to an approximation of its former size. When the window has resumed it previous proportions, release the mouse button.

## 4.3.2 Summary of olvwm Window Menu Functions

The table below lists the *olvwm* window menu functions.

| **To do this ...** | **Choose ...** |
|---|---|
| Immediately stop the window and make it disappear | Close |
| Enlarge the window to cover the entire root window | Full Size |
| Change the location of the window | Move |
| Change the size of the window | Resize |
| Send a window to the back or bottom of the window stack, the position closest to the root window | Back |
| Repaint the contents of the window | Refresh |
| Freeze the location of a window | Stick |
| Quit the X client | Quit |

## 4.3.3 olvwm Desktop Menu

The OpenLook Virtual Window Manager (*olvwm*) Desktop Menu is displayed when the third mouse button is held down in the root window or, if you haven't installed a three-button mouse, when the right arrow key is depressed. (Mouse button operations are described in the section "3.2.7 Mouse Button Mapping".)

The *olvwm* Desktop Menu can be used to invoke the X$^{\text{Ten}}$ Manager, execute local and remote X clients, access X$^{\text{Ten}}$ and *olvwm* utility functions, and hide, restart and quit X$^{\text{Ten}}$.

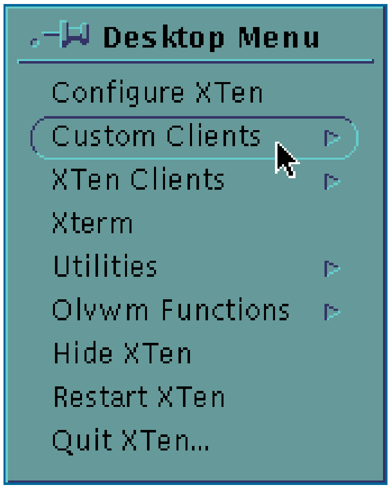


**Figure 46. olvwm Desktop Menu**

### 4.3.3.1 Configure X$^{Ten}$

The “Configure XTen” menu entry is used to start a local Web browser with parameters to access the local X$^{Ten}$ Manager.

### 4.3.3.2 Custom Clients

The “Custom Clients” menu entry is used to execute local and remote X11 clients. The X$^{Ten}$ Manager is used to define custom clients. See section “3.3 Customizing X Clients”.

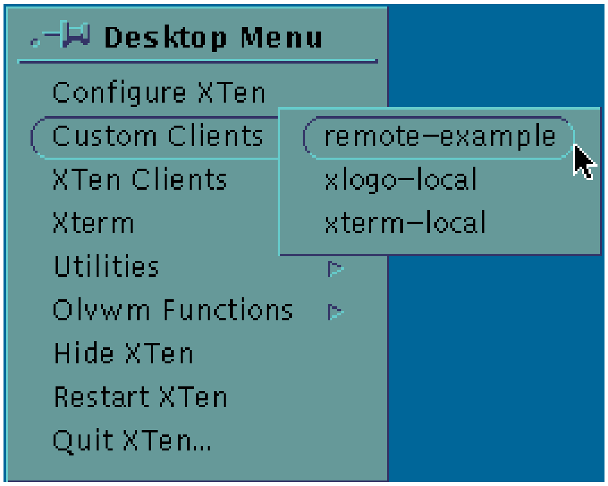


Figure 47. Custom Clients Menu

The “Custom Clients” selection show in “Figure 47. Custom Clients Menu” shows the default X$^{Ten}$ custom clients. In the example, the “remote-example” X11 client will be executed when the mouse button is released. As new custom clients are created with the X$^{Ten}$ Manager, they are displayed in this sub-menu and may be executed within this point-and-click interface.

### 4.3.3.3 X$^{Ten}$ Clients

X$^{Ten}$ includes a selection of X client applications for execution within the multi-threaded X$^{Ten}$ application. Electronic online manual information is available for each X client application. The online information may be viewed with the X$^{Ten}$ Manager or with the local *xman* X11 client. A listing of X$^{Ten}$ clients will be displayed by choosing the “XTen Clients” menu item in the Desktop Menu. Individual X clients may be launched by selecting them from this list. In the figure below, *xeyes* is the client selected to be launched when the mouse button is released.

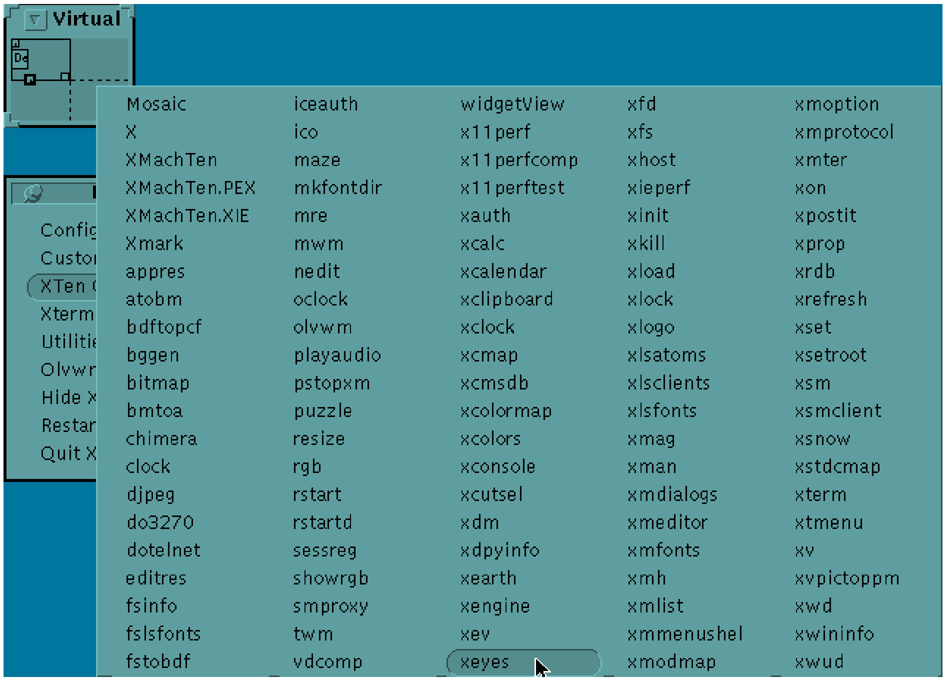


**Figure 48. X11 Clients Listing**

When the mouse button is released, *xeyes* will be launched and displayed in the root window.

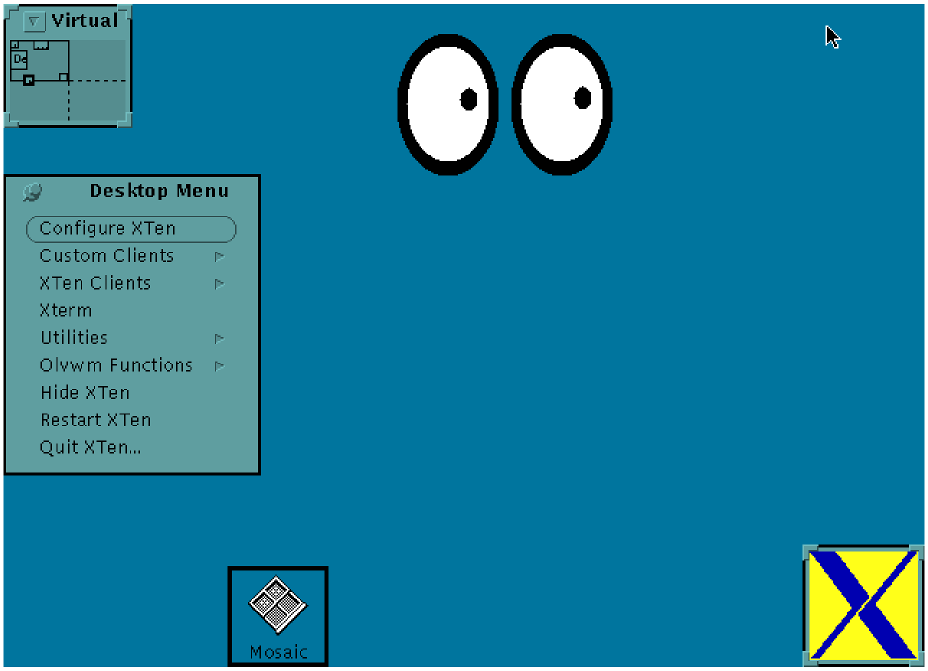


**Figure 49. xeyes Display**

### 4.3.3.4 Xterm

The "Xterm" menu entry is used to start a local xterm X11 client. Xterm is an X Window terminal window. Starting a local xterm gives you an ASCII command line environment on the X desktop. The X$^{Ten}$ xterm environment contains a limited number of commands that may be executed using a line-by-line UNIX shell command syntax. Commands are available to:

- manage local files
- create and execute local shell command scripts
- transfer data between the local system and remote systems using the *ftp* and *rcp* commands
- execute remote commands using the *rsh* and *rexec* commands
- connect to remote systems with the *telnet* and *rlogin* commands

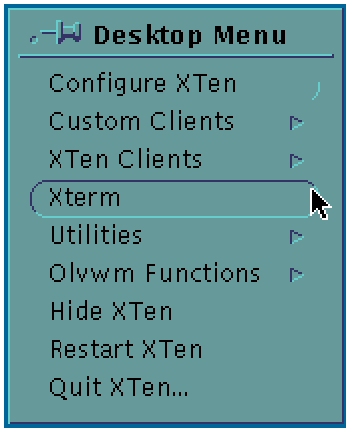
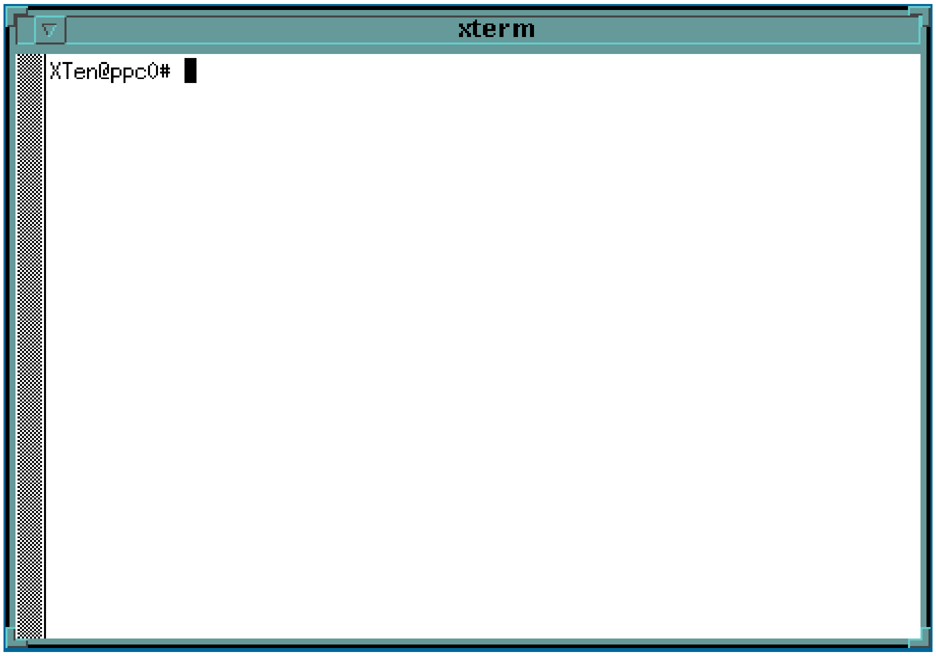


**Figure 50. Desktop Menu**

When the mouse button is released, a local X11 xterm will be run on the local root window.



```
XTen@ppc0#
```

Figure 51.  xterm Window

### 4.3.3.5 Utilities

The “Utilities” section of the Desktop Menu contains a sub-menu. The Utilities sub-menu contains entries to “lock” the Macintosh display and require a password to re-enable access to the Macintosh, an automatic startup for the *xman* electronic, online manual information browser, an image viewer implemented with the X11 *xv* client, an X11 *calendar* program, a means to define the root screen background picture, and a direct means to bring up the X11 console.

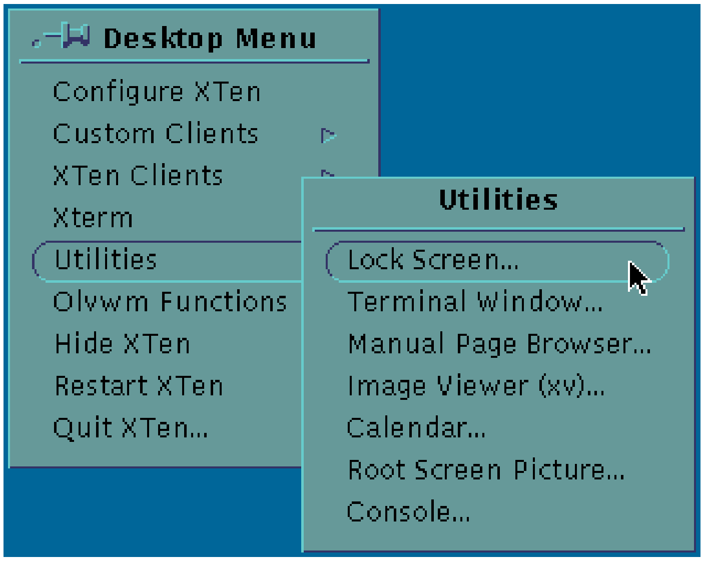


Figure 52. Utilities Menu

**Note:** Before locking the X screen, refer to “Appendix E — X^Ten Security & Passwords” for the password required to unlock the screen.

## 4.3.3.6 olvwm Functions

This menu item contains a sub-menu. The *olvwm* sub-menu contains menu items to redraw the screen, move the input focus to another window, sub-menu items to restart and quit *olvwm*, a sub-menu item to quit all X11 clients, and a sub-menu item to locate the window of a particular X11 client.

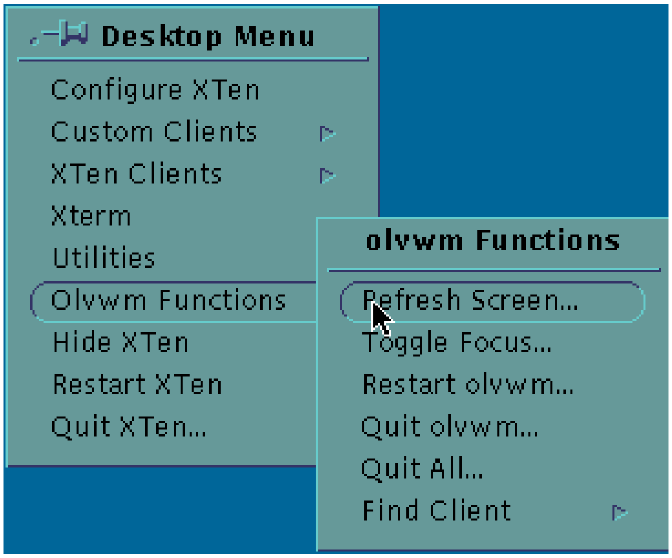


**Figure 53. olvwm Functions Menu**

The “Find Client” sub-menu has an entry for each executing X11 client. Selecting a client will cause the input focus to change to that client, and that client will be brought to the foreground.

### 4.3.3.7 Hide X$^{\text{Ten}}$

The “Hide XTen” menu item hides the X$^{\text{Ten}}$ desktop and reverts control of the display, mouse and keyboard to the Macintosh Finder. The traditional Macintosh Finder desktop is displayed and full Finder operations are restored.

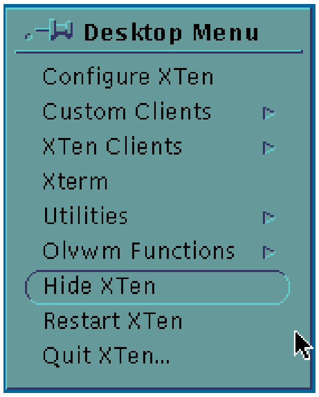


Figure 54. Desktop Menu

Access to the X$^{\text{Ten}}$ desktop may be regained by selecting the X$^{\text{Ten}}$ application from the Finder application menu in the upper right-hand corner of the Finder desktop. When X$^{\text{Ten}}$ is selected, the Finder desktop will be hidden and the X$^{\text{Ten}}$ desktop will once again take prominence.

### 4.3.3.8 Restart X$^{Ten}$

The "Restart XTen" menu item will restart the X$^{Ten}$ application. If you make a change in configuration, the X$^{Ten}$ application must be restarted to incorporate the new configuration into its operation. This menu item automatically provides for that function.

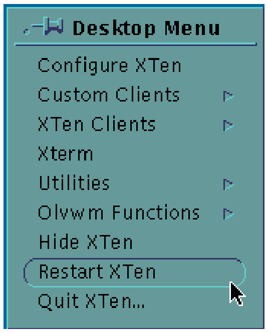


**Figure 55. Desktop Menu**

### 4.3.3.9 Quit X$^{Ten}$

The "Quit XTen" menu item will quit the X$^{Ten}$ application and return the user to the Macintosh Finder desktop.

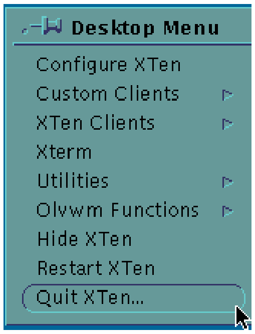
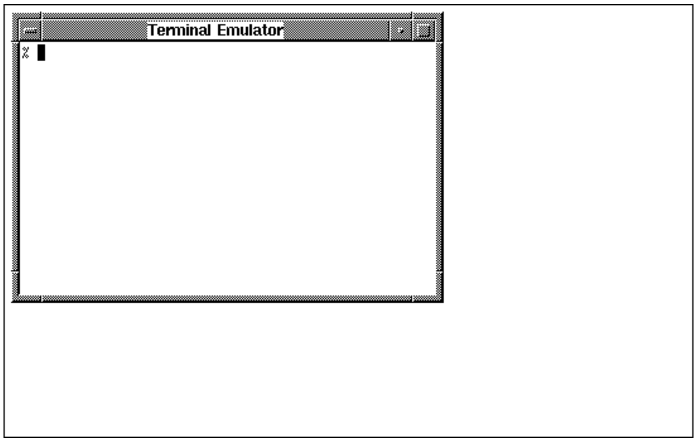


**Figure 56. Desktop Menu**

## 4.4 Navigating the Motif Desktop

Using the third mouse button, display the *mwm* root menu, select the "New Window" menu item, and start an xterm client. For information on the third mouse button, see section "3.2.7 Mouse Button Mapping". This will create an *mwm* xterm window.



```
%
```

Figure 57. Motif Environment

### 4.4.1 Motif Window Menu

In *mwm* the window menu is attached to a button in the upper left-hand corner of the window. Use the menu button in the upper left corner of each window to display the window menu. To display the *mwm* window menu, point to the menu button and then press and hold the select (mouse) button. The menu is displayed as long as you hold the button down — don't release the button yet (see below).

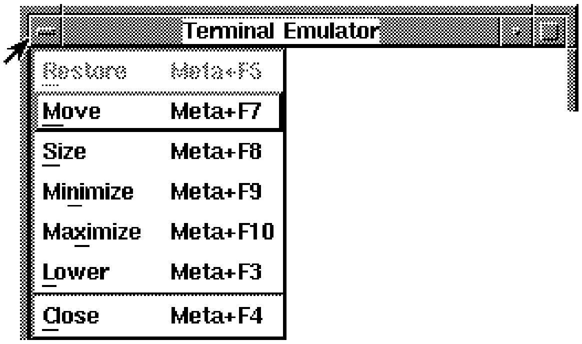


**Figure 58. Motif Window Menu**

***Step 1 — Choose a Function from the Menu***

While continuing to hold down the mouse select button, drag the pointer down the menu. As the pointer moves, it highlights a button for each available selection. Drag the pointer until you highlight the "Maximize" function. Release the mouse button.

The "Maximize" function causes the window to expand to fill the entire screen.

***Step 2 — Restore the Window to its Original Size***

Display the window menu again. Drag the pointer down the menu until you highlight the "Restore" selection. Release the mouse button. The window is restored to its original size.

### 4.4.2 Summary of Window Menu Functions

| To do this ... | Choose ... |
|---|---|
| Restore a window from an icon or after maximizing | Restore |
| Change the location of the window | Move |
| Change the size of the window | Size |
| Shrink the window to its icon representation | Minimize |
| Enlarge the window to cover the entire root window | Maximize |
| Send a window to the back or bottom of the window stack, the position closest to the root window | Lower |
| Immediately stop the window and make it disappear | Close |

When a menu function is meaningless, its name is displayed in a light shade of gray and cannot be selected.

### 4.4.3 Other Motif Title Bar Buttons

The *mwm* “Maximize” and “Minimize” menu functions also have analogs in the title bar of every *mwm* menu. Two buttons are displayed in the upper right-hand corner of each *mwm* window. The left-most of the two buttons performs a minimize function. The right-most of the two buttons performs a maximize function. Rather than choosing the “Maximize” and “Restore” functions in the previous example, you could have used the “Maximize” button in the upper right corner of the xterm window, immediately to the right of the “Minimize” button.

# 5.0 Executing Remote X Clients

X$^{Ten}$ lets you generate local requests to execute remote X clients. There is no definitive industry standard for the specification and execution of an X client from an X server. Two of the most standard methods for executing remote clients use versions of UNIX commands known as *rsh* and *rexec*. X$^{Ten}$ contains implementations of *rsh* and *rexec* which may be used to execute X clients on remote systems that support *rsh* and *rexec* service.

With X$^{Ten}$, you can execute remote X clients manually by using *rsh* and *rexec* commands within an X$^{Ten}$ xterm window, or you can use the “Custom Clients” configuration tool for smooth, menu-based initiation of X clients. When using the “Custom Clients” configuration component of the X$^{Ten}$ Manager, you will be required to specify the remote host, remote user name, and the remote command. The command syntax for the remote command must refer to valid file path names and proper command parameterization consistent with the documentation for the remote X Client.

Remote X clients can also be executed using secure connections via the secure shell (*ssh*). Like *rsh* and *rexec* commands, secure shell connections may be established manually within a local xterm window, or you can use the “Custom Clients” configuration tool for menu-based initiation of secure X clients. Before you can use secure connections you must create a security key and place it on the remote X system. See section “5.3 Remote X Client Secure Shell Example” for details.

Once a “Custom Client” is created, you can test launch the client to verify proper execution. The results of the execution are available in the “Launch Log” of the “X Client Editor”. Launching a client on a remote system requires “user” authentication. Once a “user” has been successfully authenticated, remote X client execution is straightforward. If authentication problems exist, remote UNIX systems typically respond with the string “`Permission Denied`”, and little else in the way of useful progress information.

## 5.1 Remote X Client Permissions

If permission problems persist, create a local X$^{Ten}$ xterm window and use the *rsh* command to attempt to understand the source of the permission blockage. Before you can use *rsh* or *rexec* to execute a remote X client, the remote system must be configured to accept a “user” login, and the local X$^{Ten}$ host name must be registered with the remote system as a valid host.

Make sure that the user name is a valid login on the remote system, and that the local X$^{Ten}$ host is properly identified to the remote system via an *rhosts* specification. If you do not have a valid user name and password for the remote system, you need to contact the remote system administrator for a valid remote system user name and password login.

Registering the local X$^{Ten}$ system as a valid host requires either modifying the remote system */etc/rhosts* file or adding to or creating a *.rhosts* file with the appropriate entries in the login directory of the remote system. Finally, make sure that the command string that is used to start the remote X client is valid for the remote user name. This can be verified by connecting to the remote system, logging in with the user name, and attempting to execute the X client with the command string used in the “X Client Editor” of the “Custom Client”.

Unless the remote host is properly prepared, execution of a remote command from X$^{Ten}$ will not be possible.

## 5.2 Remote X Client rsh Example

This example uses *rsh* to execute a remote client. The steps are as follows:

- create a local xterm
- log in to the remote host
- create a *.rhosts* file with host name and user information
- execute an X$^{Ten}$ *rsh* command to test access and execution

To create an xterm, pull down the *olvwm* Desktop Menu and select "xterm".

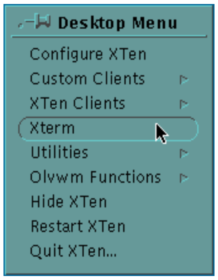
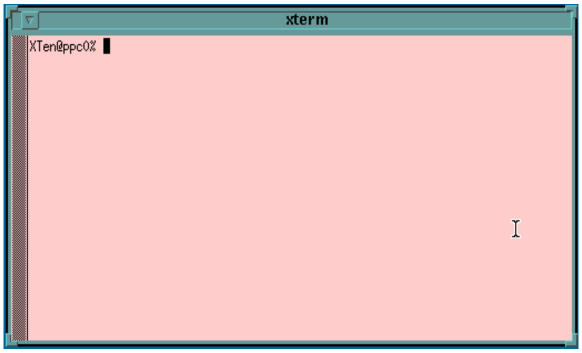


**Figure 59. Desktop Menu**

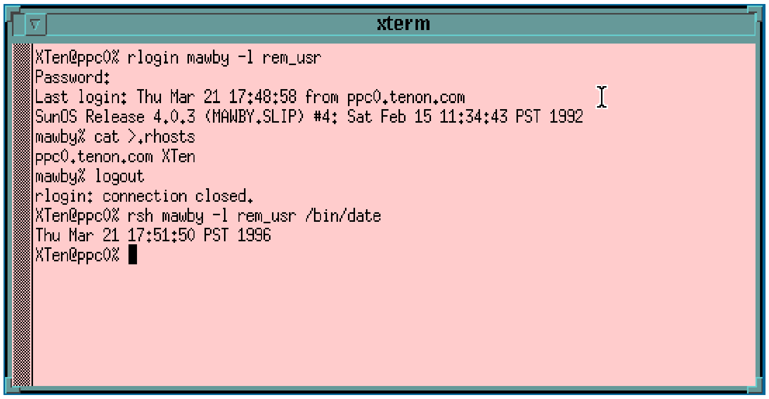
An xterm window will open with a prompt containing your X$^{Ten}$ host name.



```
XTen@ppc0%
```

**Figure 60. xterm Window**

Use the *rlogin* command at the xterm prompt to connect to the remote host using the user name and password given to you by the remote system administrator.



```
XTen@ppc0% rlogin mawby -l rem_usr
Password:
Last login: Thu Mar 21 17:48:58 from ppc0.tenon.com
SunOS Release 4.0.3 (MAWBY.SLIP) #4: Sat Feb 15 11:34:43 PST 1992
mawby% cat >.rhosts
ppc0.tenon.com XTen
mawby% logout
rlogin: connection closed.
XTen@ppc0% rsh mawby -l rem_usr /bin/date
Thu Mar 21 17:51:50 PST 1996
XTen@ppc0% █
```

**Figure 61. rlogin Session**

**Note:** The *.rhosts* file in the *rem_usr* login directory on remote host "mawby" contains an entry for the X$^{Ten}$ Macintosh host ("ppc0.tenon.com" in this example) and user ("XTen") allowed *rsh* access. Access to host "mawby" is verified by executing the */bin/date* command from *ppc0* via *rsh.*

In the example above, the *rlogin* command:

```
rlogin mawby -l rem_usr
```

connects to a remote host named "mawby" with the remote user name of "rem_usr".

To enable access from your local X$^{Ten}$ system to the remote host, create or add to the *.rhosts* file in your remote system. Add a line to the file with the completely qualified name of your local X$^{Ten}$ host name ("ppc0.tenon.com" in this example), followed by a space and the local X$^{Ten}$ login name of "`XTen`".

After typing:

```
    ppc0.tenon.com XTen
```

input a <Control>-D character, which terminates input on UNIX systems. The "mawby" prompt is printed, at which point we log out of the system. Back on the local X$^{Ten}$ system, a remote shell command to run the */bin/date* program on the remote system is created:

```
    rsh mawby -l rem_usr /bin/date
```

In the example, the remote *date* program is executed and the result is printed. This means that the local X$^{Ten}$ host has access to the remote system with the remote user name "rem_usr", and may run remote X client scripts with the same mechanism.

The X$^{Ten}$ custom X client launcher uses the *xon* utility to actually launch remote X clients. Since *xon* uses *rsh*, it is necessary for you to go through the above steps to set up and verify that the remote system permits access to your system via *rsh*. If *rsh* does not work, then the X$^{Ten}$ custom X client launcher certainly will not work.

The *xon* utility also handles other necessary tasks like setting the DISPLAY environment variable on the remote system, and enabling access to the X$^{Ten}$ X server for the remote system.

### 5.2.1 Using rlogin to Launch Remote X Clients

The tasks of enabling access to the $X^{Ten}$ X server for a particular remote system and setting the DISPLAY environment variable on the remote system can also be performed manually, enabling an *rlogin* or *telnet* session to be used as a means of launching remote X clients.

The following example uses *rlogin* to establish a connection to a remote host named “mawby”, sets the DISPLAY environment variable to point back to the $X^{Ten}$ system “ppc0”, then launches the X client xterm. Note that access from “mawby” to the X server on “ppc0” is granted (before the *rlogin* to the remote system) using the *xhost* command.

```
XTen@ppc0% xhost mawby
XTen@ppc0% rlogin mawby -l rem_user
Password:
mawby% setenv DISPLAY ppc0:0
mawby% /usr/bin/X11/xterm -geometry 80*24 -ls
```

If *rlogin* is to be used repetitively in the above fashion, the DISPLAY environment variable should be automatically set in the *.login* file on the remote system, and the command line to execute the specific X client should be entered as a script on that system.

## 5.3 Remote X Client Secure Shell Example

Secure shell (*ssh*) connections are an extension of remote shell (*rsh*) connections and, as such, they require all of the same setup steps as *rsh* connections. Refer to the previous section for information on setting up *rsh* connections. Secure shell connections also require the creation of a security key. The public part of this key must then be transferred to the remote system before secure shell connections can be established.

To generate an RSA key pair for an X$^{Ten}$ user, log in to X$^{Ten}$ as that user and issue the following command within an xterm window:

```
# ssh-keygen -N '' -b 1024 -f ~/.ssh/identity
```

Copy the public component of this key (*~/.ssh/identity.pub*) to the remote *sshd* server machine under the account you will be using to launch X clients. Add this file's contents to the *$HOME/.ssh/authorized_keys* file on the remote server.

After the public key has been copied to the remote system, secure X connections may be established from a local xterm command line. For example, use the following command to run a remote xterm client from host "test.foo.com" as user "guest":

```
# ssh test.foo.com -l guest xterm
```

The equivalent remote shell (*rsh*) command would be:

```
# rsh test.foo.com -l guest xterm -display
<your.xserver.com:0>
```

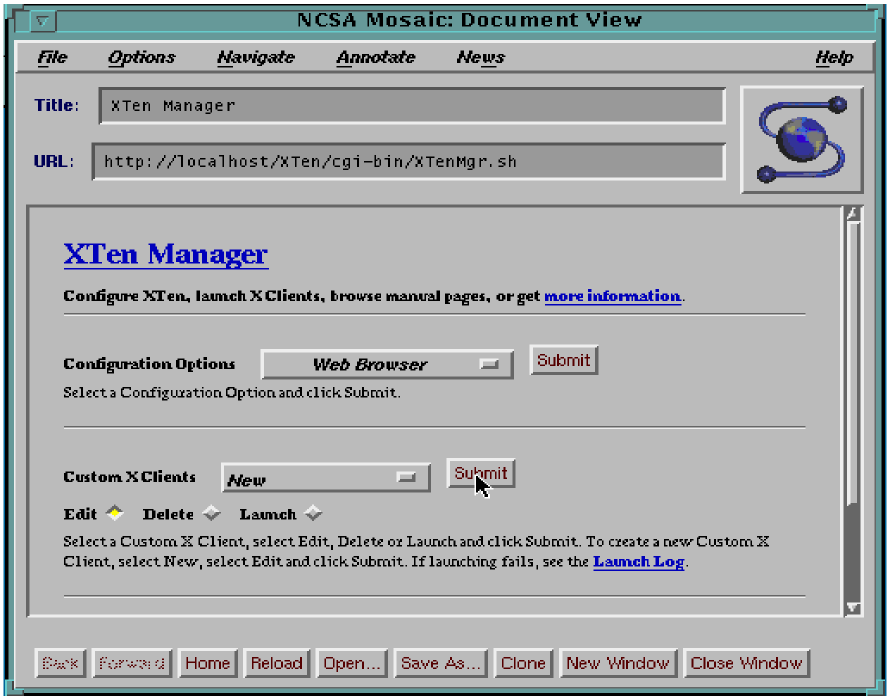
Complete *ssh* documentation is available in the online manual pages.

The remote client launch utility in the X$^{Ten}$ Configuration Manager may optionally be configured to establish an X connection using *ssh*. See section "3.3.1 Client Editor" for more information. To force the use of *ssh* for all connections established through the X$^{Ten}$ Configuration Manager (for all users), uncomment the "`#SECURESHELL=`" line in the system configuration file:

```
/usr/X11R6/lib/X11/startX/startXrc
```

## 5.4 Custom X Client Example

To create a custom X client, return to the X$^{Ten}$ Manager and select “New” in the “Custom X Clients” section.



Title: XTen Manager

URL: http://localhost/XTen/cgi-bin/XTenMgr.sh

XTen Manager

Configure XTen, launch X Clients, browse manual pages, or get more information.

Configuration Options: Web Browser

Select a Configuration Option and click Submit.

Custom X Clients: New

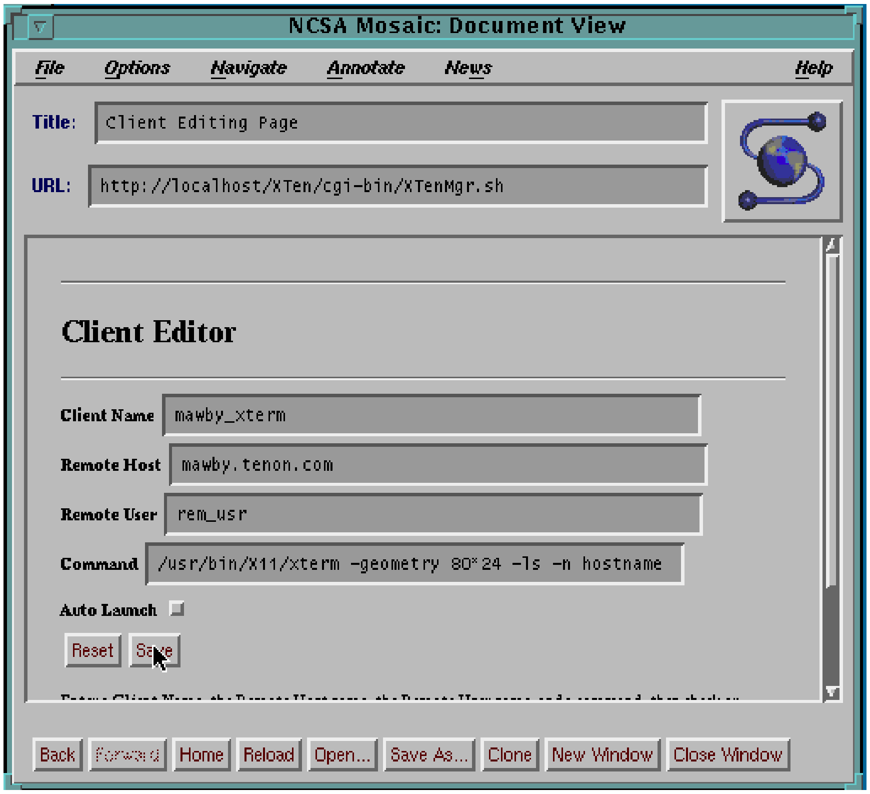
Edit ● Delete ○ Launch ○

Select a Custom X Client, select Edit, Delete or Launch and click Submit. To create a new Custom X Client, select New, select Edit and click Submit. If launching fails, see the Launch Log.

Figure 62. XTen Manager

Select the “Edit” button to create a new Custom X Client, then click “Submit”. This will display a Client Editor window with fields for the creation of a Custom X Client. Fill in the “Custom X Client” fields with the fully-qualified path name of the command, the remote host name, the remote user name, and the command to be executed on the remote host.

Then click “Save” to save the new Custom X Client in the system.



Title: Client Editing Page

URL: http://localhost/XTen/cgi-bin/XTenMgr.sh

Client Editor

Client Name: mawby_xterm

Remote Host: mawby.tenon.com

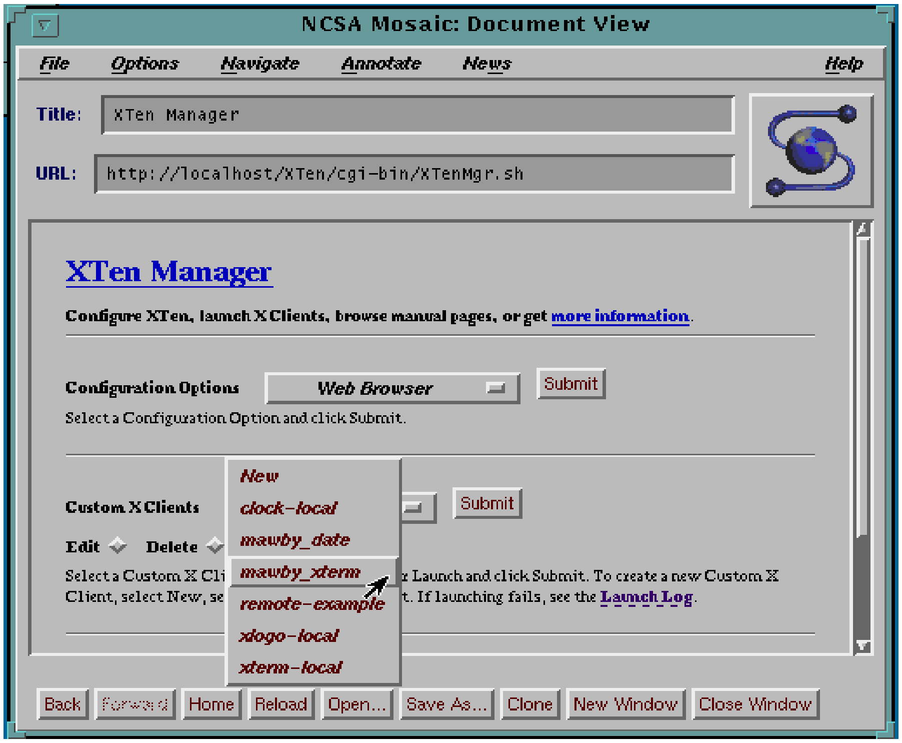
Remote User: rem_usr

Command: /usr/bin/X11/xterm -geometry 80*24 -ls -n hostname

Auto Launch □

Figure 63. Client Editor Window

After that completes, you should test launch the client. To test launch, select the new client name in the "Custom X Clients" menu ("mawby_xterm" in this case), select the "Launch" button, and click "Submit".



Title: XTen Manager

URL: http://localhost/XTen/cgi-bin/XTenMgr.sh

XTen Manager

Configure XTen, launch X Clients, browse manual pages, or get more information.

Configuration Options

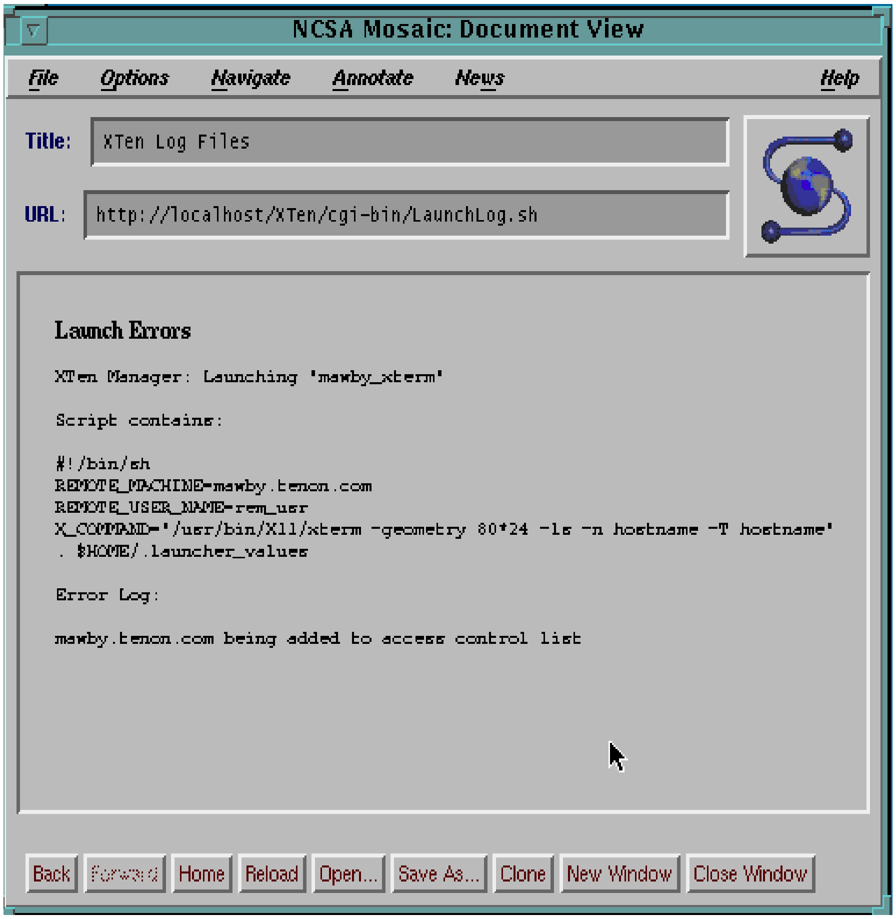
Select a Configuration Option and click Submit.

Custom X Clients

Select a Custom X Cli … Launch and click Submit. To create a new Custom X Client, select New, se … t. If launching fails, see the Launch Log.

Figure 64. Custom X Clients

This is will launch the client. You can look at the Launch Log to view intermediate error messages and any text output. Naturally, if the application is a remote X client, the results of the X client may also appear within the local X server desktop. In this case, the*/usr/bin/X11/xterm* program was executed. Note that before the execution could take place, X$^{Ten}$ automatically added the remote host to the list of hosts qualified for access to the local X server desktop.



Title: XTen Log Files

URL: http://localhost/XTen/cgi-bin/LaunchLog.sh

Launch Errors

```
XTen Manager: Launching 'mawby_xterm'

Script contains:

#!/bin/sh
REMOTE_MACHINE=mawby.tenon.com
REMOTE_USER_NAME=rem_usr
X_COMMAND='/usr/bin/X11/xterm -geometry 80*24 -ls -n hostname -T hostname'
. $HOME/.launcher_values

Error Log:

mawby.tenon.com being added to access control list
```

Figure 65.  Launch Errors

This completes the creation and test of a Custom X Client. Note that the new client will now appear in the *olvwm* Desktop Menu under "Custom Clients". Selecting the client name will cause an instance of the client to be executed on the remote host.

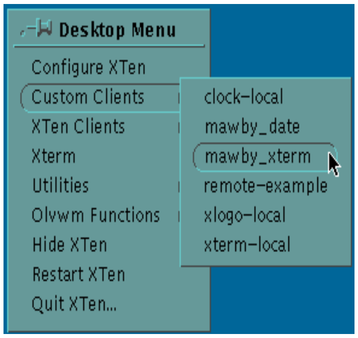
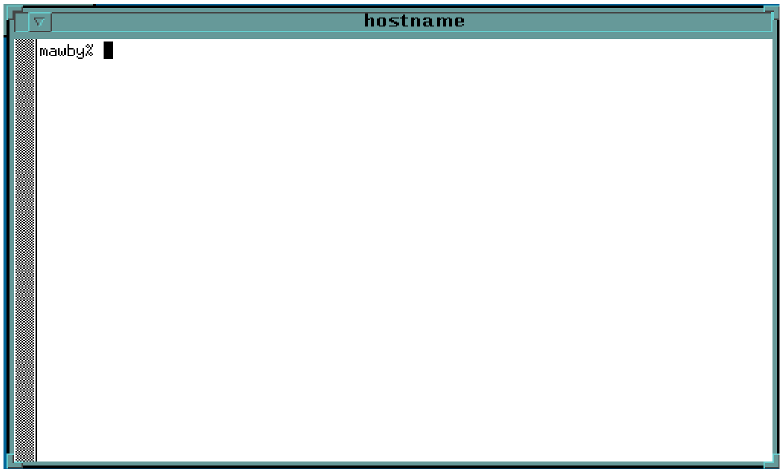


Figure 66. Custom Clients Menu

If all of the permissions are in order, the result should be an execution of the remote X client.



```
mawby%
```

Figure 67. xterm Client Window

# 6.0 X^Ten Configuration Scripting

The X^Ten software is a flexible software package that can be tailored to individual requirements. All of the configuration and maintenance that is performed via the X^Ten Manager and the X^Ten Controls control panel modify a common layer of configuration script files. This section describes those X^Ten files and the underlying X^Ten application threads that consume that configuration data. Knowledge of this underlying structure lets power X^Ten users and X^Ten system administrators make X^Ten modifications above and beyond X^Ten Manager configuration options.

## 6.1 Bootstrapping X^Ten

When X^Ten is launched (by double-clicking), the X^Ten libraries are loaded from the *Tenon Libraries* folder and X^Ten begins executing applications in the *XTen Storage* file.

**Note:** In order for X^Ten to find the *Tenon Libraries* folder and the *XTen Storage* file, they must be contained within the same folder as X^Ten.

The first application executed is *init*, which is then the ancestor of all subsequent applications. *init* runs the script */etc/rc* (see section “6.1.1 rc”), which sets up a path to the *XTen* folder and then runs the script */etc/rc.XTen*, saving its output in *XTen_HFS/etc/rc.output*.

### 6.1.1 rc

```
#!/bin/sh

SED="/usr/bin/sed"
MOUNT="/sbin/mount"
ABSOLUTE_ROOT=`$MOUNT | \
            $SED -n -e 's/^\(\/\/.*\)\/.* on \/ (.*)$/\1/p'`

# Run the XTen startup script, and save the output in a file
/bin/sh /etc/rc.XTen > "$ABSOLUTE_ROOT/XTen_HFS/etc/rc.output" 2>&1

exit 0
```

### 6.1.2 rc.XTen

The script */etc/rc.XTen* (shown below) is the script that controls the configuration of the X$^{Ten}$ execution environment. It detects and incorporates changes from the X$^{Ten}$ Controls control panel, sets the time zone, and initializes the networking interfaces.

```
#! /bin/sh -
#
# Output and error are redirected to console by init, and the console
# is the controlling terminal.

ECHO="echo"  # Shell Builtin
MV="/bin/mv"
CP="/bin/cp"
RM="/bin/rm"
LN="/bin/ln"
SED="/usr/bin/sed"
MOUNT="/sbin/mount"
DTMOUNT="/usr/bin/dtmount"
FSCK="/sbin/fsck"
GREP="/usr/bin/grep"

$ECHO ""
ABSOLUTE_ROOT=`$MOUNT | \
    $SED -n -e 's/^\(\/\/.*\)\/.* on \/ (.*)$/\1/p'`
```

```
$ECHO "Mounting Mac file systems.."
$MOUNT -va -t ufs
$ECHO "Checking Root Fast File System"
$FSCK -p "$ABSOLUTE_ROOT"/XTen_FFS
ERR_VAL=$?
if [ $ERR_VAL -ne 0 ]; then
    if [ $ERR_VAL -eq 2 ]; then
     $ECHO "Quit Requested - Running /bin/sh"
     /bin/sh
    fi
    if [ $ERR_VAL -eq 8 ]; then
     $ECHO "Returning to MacOS - NO SYNC"
     exit 1
    fi
    $ECHO "File System Check Error - Suggest checking"
    $ECHO "Fast File Systems with Fast File System First Aid"
    $ECHO "Suggest Disk First Aid for other File Systems"
    exit 1
fi

$ECHO "Mounting Root File System Read/Write"
$MOUNT -u -o rw /

#$ECHO "Mounting Root File System on Desktop"
#$DTMOUNT -o double,double_dir,.rsrc /

# Make a symbolic link to the XTen folder

$RM -f /XTen_Folder
$LN -s "$ABSOLUTE_ROOT" /XTen_Folder
$ECHO  "Linking to XTen Folder"

$ECHO "Checking Fast File Systems"
$FSCK -p
$ECHO "Mounting Fast File Systems"
$MOUNT -va -t ffs 2>&1 | $GREP -v "FFS_[1-5]00: No such file or
directory" | \
    $SED -e 's/No such file or directory/Volume not online/'

#$ECHO "Mounting Other File Systems on Desktop"
#$DTMOUNT -a -o double,double_dir,.rsrc 2>/dev/null

# Make sure boot volume is mounted for use under XTen
BOOTVOL=`/usr/sbin/sysctl -n machdep.boot_volume`
```

```
case "$BOOTVOL" in
//) # Already mounted as //, make /bootvol a symbolic link
    if [ -h /bootvol ]; then $RM /bootvol
    elif [ -d /bootvol ]; then /bin/rmdir /bootvol; fi
    if [ -e /bootvol ]; then $ECHO "Can't remove /bootvol"
    else $LN -s "$BOOTVOL" /bootvol; fi
    if [ $? -ne 0 ]; then
     $ECHO "Can't make /bootvol a symbolic link to boot volume"
    fi
    ;;
*)  # Mount boot volume on /bootvol
    if [ -h /bootvol ]; then $RM /bootvol; fi
    if [ -d /bootvol ]; then $MOUNT -v "$BOOTVOL" /bootvol
    elif [ -e /bootvol ]; then $ECHO "Can't remove non-directory
/bootvol"
    else /bin/mkdir /bootvol && $MOUNT -v "$BOOTVOL" /bootvol; fi
    if [ $? -ne 0 ]; then
     $ECHO "Can't mount boot volume '$BOOTVOL' on /bootvol"
    fi
    ;;
esac

# Get kernel version and startup info
/sbin/dmesg

if [ -s /etc/localtime ]; then
    /sbin/tzset
fi

$ECHO ""

# Apply configuration changes made via the Control Panel
CONF_FILE="/etc/Install.conf"
/usr/bin/cmp -s "/dev/preferences" "${CONF_FILE}"
if [ $? -ne 0 ]; then
    /bin/sh /etc/rc.config "/dev/preferences"
>/hfs/etc/rc.config.output 2>&1
    if [ $? -eq 0 ]; then
     # New configuration is now current
     $CP "/dev/preferences" "$CONF_FILE"
    else
     $ECHO "Unable to process /dev/preferences"
    fi
fi
```

```
/bin/sync

# Get local configuration values
. /etc/rc.conf

# Start of "BSD TCP/IP" vs "OpenTransport TCP/IP" initialization.
#
# TCP/IP networking may be provided by either the BSD TCP/IP library
# "SocketLib" or the OpenTransport TCP/IP library "OTSocketLib". The
# BSD TCP/IP is configured via the "MachTen Controls" Control Panel and
# this configuration information is passed to SocketLib in the following
# section of /etc/rc.  OpenTransport TCP/IP is configured via the
"TCP/IP"
# Control Panel and does not require (or allow) configuration in /etc/rc,
# hence this section of /etc/rc must be skipped if OTSocketLib is in use.
#
# Determine if OpenTransport TCP/IP is in use by doing an "ifconfig" on
the
# loopback interface.  If it fails (usually with "Operation not
supported"),
# OpenTransport TCP/IP is in use, so skip this section of /etc/rc.

$ECHO "Configuring loopback interface.."
/sbin/ifconfig lo0 localhost up status  2>/dev/null

if [ $? -eq 0 ]; then
     # say MT tcp configured
     $RM -fr /etc/conf/otcpconfigured
     $CP /dev/null /etc/conf/mtcpconfigured

     # Configure the network interfaces
     if [ "$IFCONFIG_ie0" ]; then
      $ECHO "Configuring Ethernet interface.."
      /sbin/ifconfig ie0 $IFCONFIG_ie0 status
     fi
     if [ "$IFCONFIG_at0" ]; then
      $ECHO "Configuring AppleTalk interface.."
      /sbin/ifconfig at0 $IFCONFIG_at0 status
     fi

     # Route packets for hosts not directly connected to my net to this
gateway.
     if [ "$GATEWAY" ]; then
      /sbin/route add default $GATEWAY 1
     fi
```

```
else
    # say OT tcp configured
    $RM -fr /etc/conf/mtcpconfigured
    $CP /dev/null /etc/conf/otcpconfigured
fi
# End of "MachTen TCP/IP" vs "OpenTransport TCP/IP" initialization.

/bin/domainname "$DOMAINNAME"
/bin/hostname "$HOSTNAME"
# Hostid is used by mactpcd when MacTCP apps request a source IP address
/bin/hostid "$HOSTNAME"

#?if [ "$PRESERVE_EDITOR_FILES" ]; then
#?  $ECHO "Preserving editor files"
#?  /usr/libexec/ex3.7preserve -a
#?fi

# Clean up "temporary" directories; /tmp is actually optional, but the
others
# should be flushed of old data which might lead to unexpected behavior
by
# applications attempting to interact with defunct daemon processes..
    $ECHO -n "Cleaning up:"
if [ "$CLEANUP_tmp" ]; then \
#       $ECHO -n " /tmp"; cd /tmp && $RM -f - * .ufs* ;\
       $ECHO -n " /tmp"; cd /tmp && $RM -f - `ls -F | egrep -v '^Desktop
|/$'` .ufs* ;\
    chmod 1777 /tmp /var/tmp ;\
fi

    $ECHO -n " /volume"; cd /volume && /bin/rmdir -- * 2>/dev/null
    $ECHO -n " /var/run"; cd /var/run && $RM -f - * .ufs*
    $ECHO -n " /etc"; cd /etc && $RM -f - *.pid .ufs*
    $ECHO -n " /dev"; cd /dev && $RM -f log printer
    $ECHO -n " /var/spool/output"; \
     cd /var/spool/output && $RM -f lpd.lock lw/lock dw/lock iw/lock
    $ECHO -n " ../mqueue"; cd /var/spool/mqueue && $RM -f nf* lf*
    $ECHO -n " ../uucp"; cd /var/spool/uucp && $RM -f LCK* STST/*
    $ECHO " ."
$RM -f /etc/nologin

/bin/sync

# Update kernel version info in "message_of_the_day"
if [ -f /etc/motd ]; then
```

```
     /sbin/version > /tmp/.motd &&\
     /usr/bin/tail +2 /etc/motd >> /tmp/.motd &&\
     $MV /tmp/.motd /etc/motd &&\
     /bin/chmod 666 /etc/motd
fi


# Pick up any local actions
# This file resides in the 'etc' folder in the 'XTen_HFS' folder.
if [ -s /hfs/etc/rc.local ]; then
      . /hfs/etc/rc.local
fi


/etc/startup_complete

/bin/sync

exit 0
```

## 6.1.3 rc.local

The script *XTen_HFS/etc/rc.local* is the final bootstrapping script. It starts the application threads that run in the background while X$^{Ten}$ is active.

```
# /etc/rc.local -- executed by /etc/rc
#

$ECHO
$ECHO "Starting local configuration:"


BASE_DIR="/hfs/etc"
PATCH_FILE="$BASE_DIR/patches/tenon.patch"

# Add patches as necessary
if [ -s "$PATCH_FILE" ]; then
     /bin/sh "$PATCH_FILE"
     if [ $? -eq 0 ]; then
         rm -f "$PATCH_FILE"
     fi
```

```
fi


# Uncomment and fill in the following lines to configure a second
Ethernet.
#
# ADDRINFO is the IP Address or hostname for the second Ethernet
interface.
# NETMASK is the corresponding Network Mask.  NETMASK is only necessary
if the
# network mask is not the default for that IP Address class. If an
explicit
# netmask is set, the setting must include the word "netmask" then a
space
# then the netmask value.

#ADDRINFO_1="hostname"
#NETMASK_1="netmask 0xffffff00"
#IFCONFIG_ie1="$ADDRINFO_1 $NETMASK_1 up"

#ADDRINFO_100="hostname"
#NETMASK_100="netmask 0xffffff00"
#IFCONFIG_ie100="$ADDRINFO_100 $NETMASK_100 up"

# If the strings above are set, configure an additional network
interface
if [ "$IFCONFIG_ie1" ]; then
     $ECHO "Configuring additional Ethernet interface
($IFCONFIG_ie1).."
     /sbin/ifconfig ie1 $IFCONFIG_ie1 status
fi

# If the strings above are set, configure a 100 Megabit network interface
if [ "$IFCONFIG_ie100" ]; then
     $ECHO "Configuring Fast Ethernet interface.."
     /sbin/ifconfig ie1 $IFCONFIG_ie100 status
fi


# Start various system daemons

if [ "$START_syslogd" -a -s /etc/syslog.conf ]; then
     $ECHO "Starting system logger"
     /usr/sbin/syslogd
fi
```

```
# Start RPC program to DARPA protocol port mapper before dependent
daemons
if [ "$START_portmap" ]; then
     $ECHO "Starting RPC port mapper"
     /usr/sbin/portmap
fi

# Support Yellow Pages service (or Network Information Service (NIS))
if [ "$START_ypbind" ]; then
     $ECHO "Starting Yellow Pages (client service)"
     /usr/sbin/ypbind
fi

if [ "$START_named" -a -s "$NAMED_BOOT" ]; then
     $ECHO "Starting Domain Name Server"
     /usr/sbin/named -b "$NAMED_BOOT"
fi

# Mount all file systems listed in the /etc/fstab file
if [ "$MOUNT_REMOTE_FILES" -a -s /etc/fstab ]; then
     $ECHO "Mounting remote filesystems.."
     /sbin/mount -va -t nfs
fi

if [ -s /etc/exports ]; then
     $ECHO "Starting NFS Server"
     /sbin/mountd
     /sbin/nfsd
fi



$ECHO -n "Starting daemons:"
if [ "$START_cron" -a \( -s /etc/crontab -o -s /etc/crontab.local \) ];
then
     $ECHO -n " cron"
     /usr/sbin/cron
fi
if [ "$START_inetd" -a -s /etc/inetd.conf ]; then
     $ECHO -n " inetd"
     /usr/sbin/inetd
fi
if [ "$START_lpd" ]; then
     $ECHO -n " lpd"
     /usr/sbin/lpd
fi
if [ "$START_macmntd" ]; then
     $ECHO -n " macmntd"
     /usr/sbin/macmntd
fi
if [ "$START_sendmail" -a -s /etc/sendmail.cf ]; then
     $ECHO -n " sendmail"
     /usr/sbin/sendmail -bd -q1h
fi
if [ "$START_postoffice" -a -s /usr/post.office/post.office ]; then
     $ECHO -n " post.office"
     /usr/post.office/post.office
fi
if [ "$START_execmgr" ]; then
     $ECHO -n " execmgr"; cd /home/root
     /usr/libexec/execmgr
fi
if [ "$START_httpd" -a -s /usr/libexec/httpd ]; then
     $ECHO -n " httpd"
     /usr/libexec/httpd
fi
if [ "$START_xdm" ]; then
     $ECHO -n " xdm"
     /usr/X11/bin/xdm
fi
$ECHO

#
# Start the X Font Server
#

if [ "$START_fs" -a -s /usr/bin/X11/fs ]; then
     $ECHO "Starting the X Font Server"
     /usr/bin/X11/fs &
fi

$ECHO "Local configuration finished."
$ECHO
```

## 6.1.4 X$^{Ten}$ Bootstrapping Picture

The following is a pictorial representation of X$^{Ten}$'s bootstrapping process.

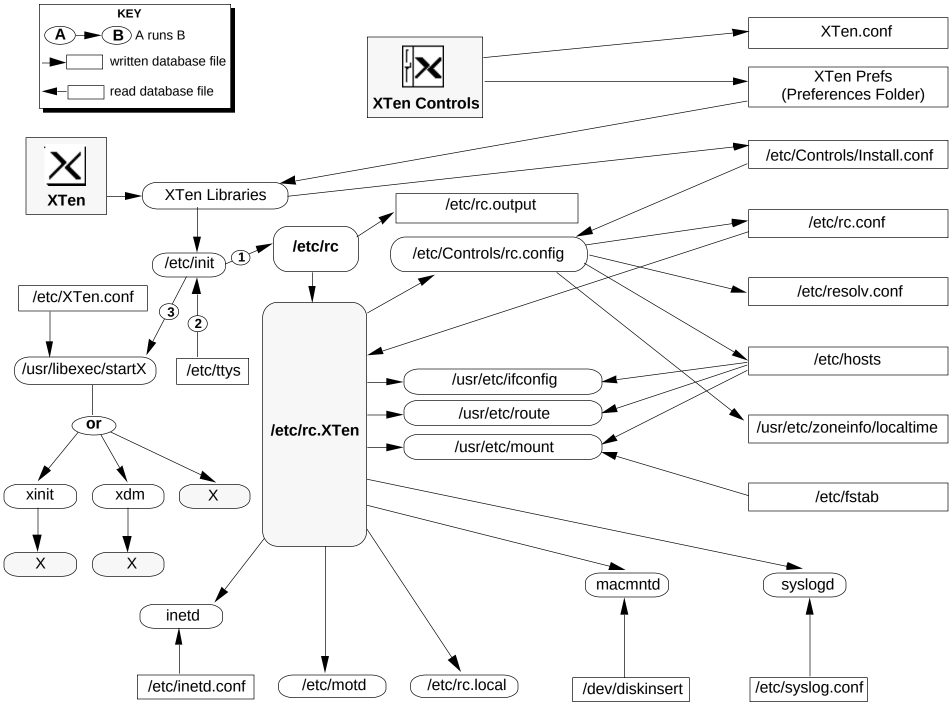


**Figure 68. X$^{Ten}$ Bootstrapping Diagram**

### 6.1.5 X$^{Ten}$ Controls and Startup Scripting

Changes made in the X$^{Ten}$ Controls control panel screens are recorded in the file */etc/Controls/Install.conf+*. Whenever */etc/rc.XTen* sees changes in this file, it invokes the shell script */etc/Controls/rc.config* to update several system files, including */etc/rc.conf*, to the new parameter values. */etc/rc.XTen* then reads the variables defined in */etc/rc.conf* and initializes the services and daemons specified.

Most variable definitions in *rc.conf* reflect the configuration settings entered in the X$^{Ten}$ Controls control panel. When you modify the control panel, the changes are applied to *rc.conf* the next time X$^{Ten}$ boots. For definitions not covered by the control panel, edit *rc.conf* and modify the variable directly. All *rc.conf* definitions are described below.

The first set of variable definitions controls the configuration of the network interface(s) installed in your system:

| | |
|---|---|
| HOSTNAME | The name of the machine running X$^{Ten}$. |
| ADDRINFO_* | The host name of the machine on Ethernet (e), AppleTalk (a), SLIP (s), and / or TokenRing (t). All assigned names should also appear in the */etc/hosts* file. |
| IFCONFIG_* | The network interface configuration parameters for Ethernet (ie0), AppleTalk (at0), SLIP (sl0), and / or TokenRing (tr0). This variable is only defined if the interface is installed in the system and is to be initialized by X$^{Ten}$. |
| NETMASK_* | The internet subnetwork mask for a configured network interface on Ethernet (e), AppleTalk (a), SLIP (s), and / or TokenRing (t). |

The following boolean variables determine whether or not an optional service is invoked, or is started by *rc*. A null value (or string) is interpreted as negative; any other value is interpreted as affirmative.

For example, the entry:

```
        START_inetd="yes"
```

will start the *inetd* application thread during the boot process, whereas the entry:

```
        START_inetd=
```

will not start *inetd*. Unused variables can be also be disabled (made negative) by preceding the entry with a pound (“#”) sign.

| | |
|---|---|
| CLEANUP_tmp | Remove files from */tmp* (aka */var/tmp*) at startup. |
| MOUNT_REMOTE_FILES | Mount all file systems recorded in */etc/fstab* at startup. |
| PRESERVE_EDITOR_FILES | Preserve *ex* and *vi* edits in progress in the event of a system crash. |
| START_cron | Execute commands by date / time via *cron*. |
| START_inetd | Service incoming internet connections via *inetd*. |
| START_macmntd | Automatically mount inserted floppies via *macmntd*. |
| START_syslogd | Log system messages via *syslogd*. By default, this variable is set if the *START_sendmail* definition is affirmative. |
| GATEWAY | The host name or internet address of the gateway to other networks. The name given should also appear in the */etc/hosts* file. The *route* command is invoked with this variable by *rc* to establish the default route. The variable should remain undefined if no gateway exists. |

When editing */etc/rc.conf*, be careful not to remove any variables which the script */etc/Controls/rc.config* expects to find.

### 6.1.6 Automatically Starting Macintosh Applications

As a final step in the startup process, */etc/rc.XTen* executes commands in a user-supplied file named */etc/rc.local*, if it exists. The */etc/rc.local* file may be used to start any machine-specific processes after X$^{Ten}$ has booted. In the following example, set the EUDORA environment variable to the actual path to the Eudora program on your system.

```
# /etc/rc.local -- executed by /etc/rc
#

EUDORA="pathname to Eudora"

($ECHO -n "Starting local services:")   >/dev/console

if [ -f "$EUDORA" ]; then
    ($ECHO -n ' Eudora')      >/dev/console
    "$EUDORA"
fi
```

### 6.1.7 Setting Up X$^{Ten}$ to Boot Automatically

X$^{Ten}$ can be set up so that it boots automatically when you start your machine. Place an alias of the *XTen* application file in the *System Folder:Startup Items* folder. Later, if you want to reset your Macintosh not to boot X$^{Ten}$ automatically, simply remove the alias files from the *Startup Items* folder.

# 6.2 X Server Startup

When *init* has finished instantiating the X$^{Ten}$ execution environment by running the */etc/rc scrip*t, it then reads the */etc/ttys* file and starts the X server by running the */usr/X11/bin/startX* script (see section “6.2.1 startX”). The *startX* script reads */XTen_HFS/etc/startX/startXrc* (see section “6.2.2 startXrc”), which controls the startup of the X server. This script reads the *.startXrc* file in the user’s home directory, which overrides settings in the system *startXrc* file.

The defaults specify that no user name or password is required to use X$^{Ten}$, and the X server will be started via *xinit*. X$^{Ten}$ can optionally be configured to require keyboard entry of a valid user name and password. See the AUTO_LOGIN setting in *startXrc*.

X$^{Ten}$ may also be started in a “server only” mode. In this mode, X$^{Ten}$ is used exclusively as an X server, which is typically the case when connecting X$^{Ten}$ to a remote system running *xdm*. See the XSERVER_ONLY and XDM_SERVER settings in *startXrc*.

## 6.2.1 startX

The following are the contents of the *startX* script:

```
#!/bin/sh
#
# Start the X window server under MachTen
#

userXconf=$HOME/.startXrc
userXoptions=$HOME/.Xparams

sysXconf=/usr/X11/lib/X11/startX/startXrc
sysXoptions=/usr/X11/lib/X11/xinit/Xparams

# Default display
DISPLAY=:0

# Read the X configuration environment.
```

```
if [ -f $userXconf ]; then
     . $userXconf
else
     if [ -f $sysXconf ]; then
         . $sysXconf
     fi
fi

# Set up custom server options

if [ -f $userXoptions ]; then
     SERVER_PARAMS="-config $userXoptions"
else
     SERVER_PARAMS="-config $sysXoptions"
fi

# Set up server X display manager (xdm) parameters

if [ -n "$XDM_METHOD" ]; then
     XDM_PARAMS="-$XDM_METHOD"
     XSERVER_ONLY="yes" ; export XSERVER_ONLY
     USE_XINIT=""
     if [ -n "$XDM_SERVER" ]; then
         XDM_PARAMS="$XDM_PARAMS $XDM_SERVER"
     fi
fi

if [ -n "$USE_XINIT" ]; then
#
# Launch X via xinit if "USE_XINIT" is set
#
     if [ -n "$I_AM_XTEN" ]; then
         if [ -n "$AUTO_LOGIN" ]; then
            USER="$AUTO_LOGIN" ; export USER
            exec /usr/X11/bin/xinit -- /usr/X11/bin/X $DISPLAY
$SERVER_PARAMS \
                > /var/adm/startXmsgs 2>&1
         else
            exec /usr/X11/bin/xdm -nodaemon \
                -server "$DISPLAY local /usr/X11/bin/X $DISPLAY
$SERVER_PARAMS" \
                > /var/adm/startXmsgs 2>&1
         fi
     else
         exec /usr/X11/bin/xinit -- /usr/X11/bin/X $DISPLAY
```

```
$SERVER_PARAMS \
            > /var/adm/startXmsgs 2>&1
     fi
else
#
# Launch X directly if "SERVER_ONLY" is set
#
     if [ -n "$XSERVER_ONLY" ]; then
         exec /usr/X11/bin/X $DISPLAY $XDM_PARAMS $SERVER_PARAMS \
            > /var/adm/startXmsgs 2>&1
     else
#
# Launch X via xdm if neither "USE_XINIT" nor "SERVER_ONLY" is set
#
         exec /usr/X11/bin/xdm -nodaemon \
            -server "$DISPLAY local /usr/X11/bin/X $DISPLAY
$SERVER_PARAMS" \
            > /var/adm/startXmsgs 2>&1
     fi
fi

exit 0
```

### 6.2.2 startXrc

The following are the contents of the */XTen_HFS/etc/startX/startXrc* file:

```
# startXrc.XTen
#
# This file contains some X configuration options.  X's startup
# script (/usr/X11/bin/startX) and other scripts source this file
# as a single point for controlling X's configuration.

# Set the global I_AM_XTEN for the rest of the X scripts.  This should
# not be changed!

I_AM_XTEN="TRUE" ; export I_AM_XTEN


# Set the display.  By default the display is set to screen 0.  To use
# another screen number, change the definition of DISPLAY accordingly.

DISPLAY=":0" ; export DISPLAY
```

```
# Use Secure Shell when launching remote clients

#SECURESHELL="/usr/bin/ssh" ; export SECURESHELL

# Start the X Server via xinit.  Using xinit ensures that the X Server is
# configured with local X Server resources (like the keyboard map).  Also,
# xinit provides a means to start either local or remote X clients.
# When USE_XINIT is defined and AUTO_LOGIN (below) is undefined, username
and
# password authentication is enforced via xdm.

USE_XINIT="yes" ; export USE_XINIT


# XTen can be configured to automatically login a specific user or require
# keyboard entry of a valid username and password.  To use automatic login,
# set AUTO_LOGIN to a valid username.  If a username other than XTen (or
root)
# is used, it must first be created using the adduser script.  To enforce
# password protected login, comment out the definition of AUTO_LOGIN, thus
# leaving it undefined. AUTO_LOGIN requires setting USE_XINIT (above).

AUTO_LOGIN="XTen" ; export AUTO_LOGIN


# X may also be started in an XSERVER_ONLY mode.  In this mode X
# is used exclusively as an X Server and all X clients execute remotely.
# This option greatly reduces X's memory size (by not loading the support
# libraries for local X clients) and it handles the common case of
connecting
# X to a remote system running xdm.
#
# Typical usage of XSERVER_ONLY includes specification of a remote xdm
server,
# XDM_SERVER (below).  It is also possible to use XSERVER_ONLY without an
xdm
# server.  In this case you probably want to set USE_XINIT (above) and edit
# "/home/$AUTO_LOGIN/.xinitrc" to explicitly launch the desired remote X
# clients.

#XSERVER_ONLY="yes" ; export XSERVER_ONLY


# To use a remote xdm server, set XDM_METHOD and XDM_SERVER accordingly.
# XDM_METHOD is the query method, one of 'query', 'indirect', or
'broadcast'.
# XDM_SERVER is the remote host running xdm, the X display manager.
# If XDM_SERVER is defined, XDM_METHOD must be either 'query' or 'indirect'.
# If XDM_METHOD is defined and XDM_SERVER is undefined,  XDM_METHOD must
# be 'broadcast'.

#XDM_METHOD="query"
#XDM_SERVER="hostname"


PATH="/usr/bin:/bin:/usr/sbin:/sbin:/usr/local/bin:/usr/X11/bin"
export PATH
```

## 6.2.3 Using an Alternate Server Display Number

The server listens for X connections on TCP port 6000+*n*, where "*n*" is the display number.  The default display number is zero (0).

X clients (run locally or from the network) access your X server by connecting to the server on the display number.  This is indicated by the DISPLAY environment variable for the shell running the X client. See the DISPLAY environment variable in the *XTen.conf* file (above):

| | |
|---|---|
| `:0` | (for local connections) |
| `<yourhostname>:0` | (for network connections) |

You can configure X$^{\text{Ten}}$ to listen for connections on an alternate display by editing the DISPLAY environment variable in the *startXrc* file. You must also change the DISPLAY variable in your local X$^{\text{Ten}}$ *.login* or *.profile* file (in the home directory of the account starting X$^{\text{Ten}}$) to match the alternate display number.

### 6.2.4 X Display Management Under X$^{Ten}$

The *xdm* program manages a collection of X displays, which may be on the local host or remote servers. The design of *xdm* was guided by the needs of X terminals as well as the X Consortium standard XDMCP, the X Display Manager Control Protocol. *xdm* provides services similar to those provided by init, getty and login on character terminals — prompting for login name and password, authenticating the user, and running a “session”. When your X$^{Ten}$ system is configured to run X$^{Ten}$ with *xdm*, you will be presented with a friendly, uniform entry into the X Window system.

*xdm* has many configuration options which are described in detail in the online *xdm* manual page. This section will concentrate on configuring X$^{Ten}$ to be controlled by *xdm*.

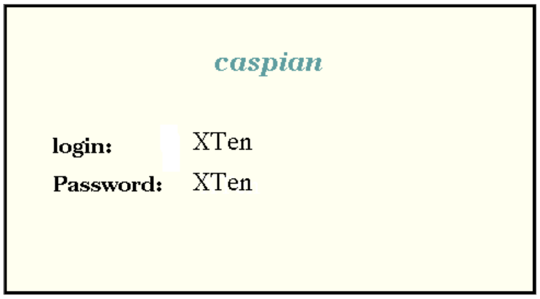
***Configuring X$^{Ten}$ to start xdm***. Use the X$^{Ten}$ Manager's “Session Control” page to configure X$^{Ten}$ for “Local XDM”, “Remote XDM” or “Chooser XDM”. Quit the X$^{Ten}$ session to restart X$^{Ten}$ under the control of the selected XDM option.

***XDM User Accounts.*** The *xdm* login window prompts for a user “account” consisting of a user name and password. You can create multiple user accounts under X$^{Ten}$ in the X$^{Ten}$ Manager. The default user name and password are both “XTen”.

***The XDM Session***. When you log into the *xdm* login window, the user session is started. The session is controlled by an executable shell script in your home directory called *.xtenrc*. If this file is not found, *xdm* runs a default session from the file */usr/lib/X11/xdm/Xsession*. When the last client (typically the window manager client) in the *.xtenrc* file exits, the session completes and *xdm* presents a new login window.

### 6.2.4.1 The XDM Login Window

Using *xdm*, X$^{Ten}$ will present a login window for security authentication when it is launched. The name of your Macintosh contained in the X$^{Ten}$ Controls control panel will appear in the window title. The initial login and password are both "XTen". See "Appendix E — X$^{Ten}$ Security & Passwords" for an explanation on how to customize your password.



caspian

login: XTen
Password: XTen

**Figure 69. The XDM Login Window**

### 6.2.4.2 Remote XDM Sessions

When launched, X$^{Ten}$ can connect to a remote *xdm* program to control the user session to use the X$^{Ten}$ Manager. To configure X$^{Ten}$ to be controlled by a remote *xdm*, see section "3.2.1 Controlling XDM Sessions".

## 6.3 Starting Clients

The initial X clients you choose to run on your X desktop are controlled by a shell script file in your X$^{Ten}$ login directory called *.xinitrc*. If *xinit* cannot find a *.xinitrc* file in your home directory, the shell script */usr/lib/X11/xinit/xinitrc* is run instead. Commented lines in this file (those with a preceding pound ("#") sign) are not executed by *xinit*.

The default X client startup script */home/XTen/.xinitrc* is shown below.

```
#!/bin/sh
userresources=$HOME/.Xresources
usermodmap=$HOME/.Xmodmap
sysresources=/usr/X11/lib/X11/xinit/.Xresources
sysmodmap=/usr/X11/lib/X11/xinit/.Xmodmap

# merge in defaults and keymaps

if [ -f $sysresources ]; then
    /usr/bin/X11/xrdb -merge $sysresources
fi

if [ -f $sysmodmap ]; then
    /usr/bin/X11/xmodmap $sysmodmap
fi

if [ -f $userresources ]; then
    /usr/bin/X11/xrdb -merge $userresources
fi

if [ -f $usermodmap ]; then
    /usr/bin/X11/xmodmap $usermodmap
fi

# Allow local connections
/usr/bin/X11/xhost `hostname`

if [ -n "$XSERVER_ONLY" ]; then

# To use the SERVER_ONLY option without an xdm server it is likely
      necessary
# to start at least one X client on a remote machine.  Xon is
      probably the
```

```
# easiest way to start such a client, but rexec and rsh may also
       be used.
# Some examples are shown below.

#xon <remote machine> -user <remote user name> <remote command>  &
#xon ahost.com -user john xterm  &

else

# Start up the Web Browser if User selected it for autolaunch.
AUTOLAUNCH_BROWSER="yes"
if [ -n "$AUTOLAUNCH_BROWSER" ]; then
./.browser
fi

# Start the User's Custom Client Scripts that are selected for
       autolaunch
./.client_autolaunch

#WINDOW_MGR=olvwm
#WINDOW_MGR=twm
#WINDOW_MGR=mwm
#WINDOW_MGR=afterstep
WINDOW_MGR=afterstep

if [ -x /usr/X11/bin/$WINDOW_MGR ]; then
  if [ $WINDOW_MGR = "afterstep" ]; then
   LOW_COLOR=`xdpyinfo | grep  "depth of root window:   8 planes"`
    if [ -n "$LOW_COLOR" ]; then
      if [ -r "$HOME"/.steprc.8bit ]; then exec afterstep -f
       "$HOME"/.steprc.8bit
        else exec afterstep -f
       /usr/X11/lib/X11/afterstep/system.steprc.8bit
      fi
    fi
  fi
  exec /usr/bin/X11/$WINDOW_MGR
else
  exec /usr/bin/X11/xterm
fi
fi
```

### 6.3.1 Initializing X Resources

The startup script begins by calling the *xrdb* client, which reads in any system-wide resources defined in the file */usr/lib/X11/xinit/.Xresources*, if present. Resources that you want available to all users and all clients displaying to the server should be defined in this file. Resources are discussed in more detail in “Appendix C — Resources: X Application Preferences”.

Next the script runs the *xmodmap* client to load a custom keyboard mapping file from */usr/lib/X11/xinit/.Xmodmap*, if present. If you are using a U.S. keyboard, you will normally have no need to change your keyboard mapping since the X server is launched by default with a U.S. keyboard mapping. If you operate the X server using a non-U.S. keyboard, the *.Xmodmap* file should be linked to one of the international keyboard map files in */usr/lib/X11/xinit*. Details on using the international keyboard map files are found in section “3.2.4 Selecting a Keyboard Mapping”.

The startup script runs *xrdb* and *xmodmap* again using the *.Xresources* and *.Xmodmap* files from your home directory. These files can be maintained in individual user home directories to customize the default server resources or key map for that user.

## 6.3.2 Initializing the X Desktop

Next *.xinitrc* checks if the XSERVER_ONLY mode of operation was specified, and refrains from starting any local X clients if it was. Otherwise, it continues by checking if the Web browser should be launched on startup, starting it accordingly.

Next the collection of custom X clients built using the X$^{Ten}$ Manager's Client Editor (and specified for "auto launch") is launched by running the *.client_autolaunch* script. The default configuration of *.client_autolaunch* "auto launches" the *xlogo-local* example.

Finally, the startup script launches a special X application known as the "window manager", which allows you to move, resize and iconify your X application windows. By default, the OpenLook Virtual Window Manager (*olvwm*) is launched. Also available in X$^{Ten}$ is the AfterStep Window Manager (*afterstep*), the Motif Window Manager (*mwm*), and the Tab Window Manager (*twm*), from the MIT sources distributed with the X Window software package.

## 6.4 Controlling X Client Access to the Server

In its default mode of operation, X$^{Ten}$ uses a host-based access control list for deciding whether or not to accept connections from clients on a particular machine. If no other authorization mechanism is being used, this list initially consists of the host on which the server is running, as well as any machines listed in the file */etc/X0.hosts*. Each line of the file should contain an internet dot address. There should be no leading or trailing spaces on any lines. For example:

```
192.84.6.5
128.111.14.4
```

You can add or remove hosts from this list, and enable or disable access control using the *xhost* command from the same machine as the server.

# 7.0 X Over Serial Lines

Under X$^{Ten}$, you may run X client applications over serial lines to display on your local X server using the Serial Line Internet Protocol (SLIP), the Point-to-Point Protocol (PPP) or, with the aid of a remote AppleTalk-to-Internet router, Apple Remote Access.

The *-wm* backing store X server startup option is a potentially useful option when displaying X clients over slow serial lines. This option causes the server to save a local copy of a window's contents when the window becomes obscured by an overlapping window. As windows are rearranged and the obscured window's contents become exposed, the window is instantly redrawn by the X server. With the option disabled, the remote X client must retransmit the window's contents over the serial line when the window becomes exposed.

Place the *-wm* backing store option in the X server startup options file */usr/lib/X11/xinit/Xparams*.

## 7.1 Using PPP

The Point-to-Point Protocol was developed to support serial communications in a multi-protocol environment. SLIP, as its name implies, is specifically for IP communications. PPP, on the other hand, as described in RFC 1331, "The Point-to-Point Protocol for the Transmission of Multi-Protocol Datagram Links", is designed to encapsulate protocols other than just TCP / IP. Because of its enhanced features, including password authentication, PPP is rapidly becoming the standard for serial communications by Internet Service Providers.

## 7.1.1 Setting Up $X^{Ten}$ PPP on Your Macintosh

This section describes how to configure your $X^{Ten}$ system using the Macintosh modem port to connect to remote PPP servers (as a PPP client).

## 7.1.2 Configuring Your $X^{Ten}$ System as a PPP Client

The recommended procedure for using $X^{Ten}$ as a PPP client is to switch to the OpenTransport Option (see “Appendix G — Using OpenTransport or Replacing OpenTransport”). Using OpenTransport enables the use of any Macintosh OpenTransport PPP client (such as FreePPP) to connect to your Internet Service Provider. Once you are using OpenTransport, you do not need to do anything special in $X^{Ten}$ to set up PPP. $X^{Ten}$ will use PPP automatically through OpenTransport.

If you do not have OpenTransport, you will need to set up PPP from within $X^{Ten}$. The remainder of this section describes how to do this.

The first step in setting up a PPP client is to connect a modem to your Macintosh. The recommended procedure is to use the Communications ToolBox to handle the modem configuration. Test the modem connection by using the *tip* command to dial out:

```
tip cm.out
```

If you wish to access the serial ports directly, the Macintosh “modem” port is “ttya” and the “printer” port is “ttyb”. The *tip* commands are:

```
tip ttya
```

or

```
tip ttyb
```

$X^{Ten}$ supports hardware flow control for the devices “ttyfa” and “ttyfb”. Note that hardware flow control requires a special cable.

If you are using the Communications ToolBox, using the “cm.out” line will cause a Communications ToolBox dialog to appear.

- Adjust the settings for your modem and dial out.

  You should then see your remote system's prompt.

- **IMPORTANT!** While you have your remote system's prompt, pull down the X$^{Ten}$ File menu and select the last item in the menu.

  You will be presented with a standard "Save" dialog.

Save *cm.out1* into your *dev* folder. Use *cm.out2*, *cm.out3*, etc., for dial-outs to other phone numbers. To exit the *tip* program, press the <return>-<~><.> keys, and then press <return> again.

The file */etc/remote* controls which lines are accessible to the *tip* command. The first few lines already have entries. You will have to add more entries if you have more dial-outs. The */etc/pppclient* script does not use */etc/remote.*

Once dial-out access has been established, it is time to edit the */etc/pppclient* script. You will probably want to print out this script and study it a bit before making any changes. It contains complete instructions. In a nutshell, you must set your dial-out line, your login account and your password. If you run the *pppclient* script with the "debug" flag, the *pppd* program will show more information when the script runs.

You should use the *tip* command to manually log in to your PPP server so that you can take exact notes on the interaction before trying to compose the *chat* script.

To terminate a PPP connection from X$^{Ten}$, first determine the process number for the *pppd* program by typing the following command:

```
ps -ax
```

Terminate the *pppd* process by typing the following:

```
kill <pid>
```

where "`<pid>`" is the process number of the *pppd* process.

Running the */etc/pppclient* script when a PPP connection is already in place will return status information about the connection.

Note that */etc/pppclient* can be run from */etc/rc.local* to start up a connection at boot time. It can also be run from *crontab* to ensure that connections are re-established if the line drops.

The default configuration is for the PPP server to supply the client with a dynamic address. There is a CLIENT variable in */etc/pppclient* which can be set to a static address for the client.

# 8.0 Advanced Features of X$^{Ten}$

## 8.1 Providing Fonts Over the Network

The *xfs* program is a font server that runs on your X$^{Ten}$ system and provides fonts to other network-based X terminals and servers.

*xfs* has many configuration options, which are described in detail in the online manual page. In most environments, *xfs* will run using the default configuration.

Perform the following steps to run the font server:

- Make certain you have installed the “Complete” package from your X$^{Ten}$ distribution.

- As the *root* user, start the *xfs* program:

```
/usr/bin/X11/xfs &
```

- To start the font server automatically when X$^{Ten}$ boots, add an entry to your */etc/rc.local* file:

```
if [ -x /usr/bin/X11/xfs ]; then
   ($ECHO "Starting the font server")    >/dev/console
   /usr/bin/X11/xfs &
fi
```

By default, *xfs* listens on TCP port 7100 for remote font service requests. For compatibility with X11R5-based X terminals and servers requesting font service, you may optionally start *xfs* to listen on TCP port 7000:

```
/usr/bin/X11/xfs -port 7000 &
```

# APPENDIX A

## Optimizing X$^{Ten}$

You can optimize X$^{Ten}$ performance by tailoring your system to fit your particular hardware configuration and software service needs. X$^{Ten}$ performance can be dramatically improved by giving some attention to the following configuration options.

- **Use a fast disk driver**

  The efficiency of the disk device driver plays an important role in the overall performance of the system. Choosing a fast disk and disk driver (such as FWB[1]) and adhering to the guidelines of disk optimizing tools will improve the X$^{Ten}$ system's performance.

- **Use disk caching**

  Setting the disk cache in the Macintosh Memory control panel is essential to getting the most performance from your Macintosh. MacOS file manipulation software operates much more efficiently when disk caching is enabled and set to a large value. In some cases, the larger the value the better. A disk cache size of 512K is recommended; improved performance gains under X$^{Ten}$ tend to diminish beyond this size.

- **Turn off Apple File Sharing**

  Disable the file sharing option located in the Sharing Setup control panel.

[1] FWB, Inc., 2040 Polk Street, Suite 215, San Francisco, CA 9109, 415-474-8055.

- **Close the control panels after use**

  Do not keep the X$^{Ten}$ Scheduling Priority slide bar or any control panel on the desktop. An open control panel consumes valuable CPU cycles.[1]

- **Check color depth in the Monitors control pan**el

  For best performance, match your system's color settings to the requirements of your X clients. If, for example, your X clients require only 256 colors, configure the Monitors control panel accordingly.

[1] Refer to section "2.2 X$^{Ten}$ Control Panel" for more details on using the slide bar.

# APPENDIX B
## Internet (IP) Addresses

The Networking Configuration screen is a convenient way to configure the primary interface on AppleTalk and Ethernet for internet networking (see section "2.4.2 Networking Configuration Screen"). For detailed instructions on how to configure PPP interfaces, see section, "7.1 Using PPP". Each configured interface (described below) must be assigned a unique internet address in standard internet "dot" (decimal) format. The two-letter description in parentheses is a short version of the interface name as used by X$^{Ten}$. It is followed by a zero, indicating the primary interface.

***AppleTalk (at0).*** If your Macintosh is connected to a LocalTalk network or your Apple TCP / IP control panel is configured for AppleTalk over Ethernet ("EtherTalk"), assign an internet address to the AppleTalk interface. The default value, if shown, comes from your TCP / IP configuration. You will be able to use Apple networking facilities (printers, file sharing, etc.) without assigning an internet address to your LocalTalk interface; however, if you wish to have IP connectivity on AppleTalk, it must have an internet address. (This is useful if, for example, users with NCSA Telnet on AppleTalk want to connect to your X$^{Ten}$ Macintosh directly over AppleTalk.) If a MacIP gateway and dynamic addressing are used on your network, enter an IP address of "0.0.0.0".[1] (Unless you plan to use your X$^{Ten}$ networking code solely for Telnet access to some other system, it is recommended that you do not use dynamic internet addressing.)

In a system with more than one type of networking interface, the host name assigned by X$^{Ten}$ to the AppleTalk interface will be "host_a", where *host* is the host name selected in the General Configuration screen. If this interface is the only configured interface, the host name associated with the IP address entered will simply be "host".

[1] By default, X$^{Ten}$ scans the locally connected AppleTalk zone for an IP gateway that will provide the IP address. X$^{Ten}$ may be configured to scan multiple zones to locate a MacIP gateway during dynamic IP address assignment. See the *Configuration Resources* technical note in the */pub/tech_notes* directory on *ftp.tenon.com*.

Users running EtherTalk on their system may elect to configure the corresponding Ethernet interface under X$^{Ten}$ in addition to the AppleTalk interface. This requires the assignment of a unique IP address (that is, different than the address assigned to AppleTalk) to the Ethernet interface described below. In this configuration on Ethernet, users would have direct access to other non-Macintosh internet hosts from X$^{Ten}$, without the overhead of EtherTalk packet conversion by an AppleTalk-to-Internet gateway, while still enjoying connectivity with AppleTalk hosts via EtherTalk.

In effect, two logically distinct network nodes are created on your Macintosh which share a common Ethernet cable to send and receive packets. The host name on EtherTalk assigned by X$^{Ten}$ and corresponding to the AppleTalk IP address entered would be “host_a”, and the Ethernet host name corresponding to the Ethernet IP address entered would be “host” with an alias of “host_e”, where *host* is the host name selected in the General Configuration screen of the X$^{Ten}$ Controls control panel.

***Ethernet (ie0).*** If your Macintosh is connected to an Ethernet, this field represents the assigned internet address of the Ethernet network interface. The IP address from the Apple TCP/IP control panel will appear by default if TCP/IP has been configured for communication over Ethernet.

In a multiple interface configuration, the host name assigned by X$^{Ten}$ to this interface will be “host”, where *host* is the host name selected in the General Configuration screen. The host name alias “host_e” will also be assigned to this interface.

***Default Gateway.*** This field represents the internet address in standard internet “dot” format of the default gateway system on your network, and defaults to the “gateway” field in the Apple TCP/IP control panel. All access to systems not on your network will be made through the default gateway.

***Domain Name Server and Domain Name.*** The “Domain Name Server” field represents the internet address in internet “dot” format of the network domain name server. The domain name is an ASCII string (for example, *tenon.com*) representing your network domain name. Both entries default to the “Default” Domain Name Server IP Address and Domain in the Apple TCP/IP control panel.

## Netmasks

If subnetworking is implemented on your network, the "Netmask" fields are used to derive the network and host (node) portions of the internet address for a given network interface.

Netmasks are entered either in standard internet "dot" (decimal) format or in hexadecimal format preceded by the characters '0x'. The default value is taken from the Apple TCP / IP control panel. If subnetworking is not employed on any of your networks, the "Netmask" field for each configured interface should be left blank. In this case, X$^{Ten}$ will determine the correct netmask based on the interface IP address.

# APPENDIX C
## Resources: X Application Preferences

Macintosh applications permit the setting of preferences to customize the application to the individual user's tastes; for example background color, window title, font size, etc. Preferences are set for the application and are typically stored in a single file in the Macintosh Preferences folder.

Under X, preferences are set via application "resources". Resources are specified as strings that are read in from multiple source files when an application is run. Application resources under X are inherently more flexible than Macintosh application preferences due to the client / server model under which X operates. As with Macintosh programs, preferences can be specified for the individual X application. In addition, X application preferences can also be specified for each individual user. Since the same resource source directory may be shared by several different hosts (for example, an NFS file server serving a user's login directory to multiple hosts), the user's application preferences can be specified for the specific host. Similarly, because X clients can display on any networked X server and each server may need special preferences, resources can be defined specific to X$^{Ten}$.

***Resource Files.*** Resource files contain the default application resources and are commonly provided with the X application you are running. The resource file is named after the "class" name of the application. By convention, the class name of the application is the same as the program name, but with the first letter capitalized (for example Bitmap or Emacs), although some programs that begin with the letter "x" also capitalize the second letter. Application class names can be discovered using the *xprop* utility.

***Resource Syntax.*** In their simplest form, resources have the following syntax:

```
name*variable:value
```

For example:

```
XTerm*background:beige
```

where *name* is either the class name or the “instance” name of the application. The default instance name is the name of the program, for example:

```
xterm*background:blue
```

Instance names may be changed using the -*name* command line option when you start the application. You can use the -*name* option to distinguish more than one instance of the same application in your resource file.

As an example, suppose you want xterm windows from host “ray” to have a blue background and xterm windows from host “fred” to have a red background. Your resource file may contain the following resources:

```
xterm-ray*background:blue
xterm-fred*background:red
```

When you start the xterm client from hosts “ray” and “fred”, use the -*name* option to match the instance name used in the resource:

```
xterm -name xterm-ray &      (xterm started from “ray”)
xterm -name xterm-fred &     (xterm started from “fred”)
```

“Instance” names take precedence over “class” names in resource specifications.

Applications written with the X Toolkit Intrinsics will have at least the following resource variables:

background This resource specifies the color to use for the window background.

foreground This resource specifies the color to use for the text and graphics within the window.

borderWidth This resource specifies the width in pixels of the window border.

borderColor This resource specifies the color to use for the window border.

***Resource File Locations.*** Programs based on the X Toolkit Intrinsics obtain resources from the following sources. The sources are listed in the order of precedence from lowest to highest precedence; if the same resource specification is located in more than one place, the resource in the source with the higher precedence is used.

### *Application Level Resources.*

- **/usr/lib/X11/app-defaults**

  This directory is akin to the Macintosh Preferences folder. This is a directory of default application resource files. Within each file, for example *XTerm*, resource variables are defined for the application. When started, the application consults only the file in this directory corresponding to the “class” name of the application.

### *User Level Resources.*

- **$HOME/<application file>**

  The application default resource file may be copied to your home directory and customized according to your individual preferences. The file *~/XTerm* will take precedence over */usr/lib/X11/app-defaults/XTerm* when *xterm* is started.

- **XAPPLRESDIR**

  This is an environment variable you can set to an alternate search path for your application resource files; for example:

  ```
  setenv XAPPLRESDIR /tmp/app-defaults
  ```

  causes the X client to search for its resource file from an *app-defaults* sub-directory under the */tmp* directory.

- **$HOME/.Xdefaults**

  If the X$^{Ten}$ database (described below) contains no resources, resources are read from a *.Xdefaults* file in your home directory when the application starts up. This file is not specific to an application and may contain resources from any number of different X clients.

### *X Server Level Resources.*

- **X Server Database**

  Any global resources that should be available to all clients on all machines can be stored in a database residing on X$^{Ten}$ using the *xrdb* program. This is typically done when the server starts up in the *.xinitrc* file in the user's home directory. If the file *usr/lib/X11/xinit/.Xresources* containing server level resources is found, the resources are loaded into X$^{Ten}$ and automatically become available to all client applications from all systems, including the local host system. The X$^{Ten}$ database is loaded using the command:

  ```
  xrdb -merge <resource file>
  ```

  If any resource is loaded into the X$^{Ten}$ database via *xrdb* (including non-conflicting resources), the *$HOME/.Xdefaults* file will not be read.

### *Host System Level Resources.*

- **$HOME/.Xdefaults-<hostname>**

  This file is associated with the host (from the *hostname* command) on which you are executing the X application. It is useful when you share a single NFS-mounted home directory among multiple hosts. It takes precedence over all application, user, and server level resources previously mentioned. It is not specific to an application and may contain resources for any number of different X applications.

- **XENVIRONMENT**

  If this environment variable is set to a named file, the file is read instead of the *.Xdefaults-<hostname>* file. It takes precedence over all other previously mentioned resource locations when starting an X application. For example:

  ```
  setenv XENVIRONMENT ~/.Xdefaults-custom
  ```

  The named file typically contains custom resources for all of the applications you run.

***Command Line Resource Specification.***

- **-xrm *resourcestring***

  Resources can also be specified from the command line when starting your X application. The *resourcestring* is a single resource name, variable and value.

  For example, the *xterm* default background color is overridden with a red background using the command:

  ```
  xterm -xrm "XTerm*VT100.background: red"
  ```

  Note that if the string contains characters interpreted by the shell (e.g., an asterisk), they must be contained in double quotes. Any number of *-xrm* arguments may be given on the command line.

- **Command Line Options**

  A command line option to the X application takes ultimate precedence over all other resource locations. An *xterm*'s background will be blue regardless of all *xterm* background resource specifications in all locations when *xterm* is started with the command:

  ```
  xterm -bg blue
  ```

***Recommendations for Administering Resources***. Maintain unmodified copies of application resources in the */usr/lib/X11/app-defaults* directory. Customize these resources for individual users by copying the customized variables from each resource file to a *.Xdefaults* file in the user's home directory.

If you run the same client from multiple hosts and you want to retain a consistent appearance for the client, consider using *xrdb* to load the client's resources directly into X$^{Ten}$. This method eliminates the need to maintain multiple identical copies of your *.Xdefaults* file on each host.

If in doubt as to which resources an X application will use when it is launched, use the *appres* program to preview the application's resources.

For more information on the resource variables available to an application, refer to the online manual page for the application.

# APPENDIX D
## The X$^{Ten}$ Program

When X$^{Ten}$ starts up, it takes over the display and conditionally takes over the Macintosh menu bar.

The X$^{Ten}$ application is the file */usr/bin/X11/XMachTen* or, by its alias name, */usr/bin/X11/X*. By default, *xdm* and *xinit* will automatically run */usr/bin/X11/X*.

### X$^{Ten}$ Startup Options

Use the X$^{Ten}$ configuration page to set the X$^{Ten}$ startup options. The configuration page will save the selected options to the *./usr/lib/X11/xinit/Xparams* file. The complete option list is shown below:

| | |
|---|---|
| ac | Disables host-based access control mechanisms. Enables access by any host, and permits any host to modify the access control list. Use with extreme caution. This option exists primarily for running test suites remotely. |
| -apple *alt* | Defines the Macintosh <Command> key as an X “alt” key. By default, the <Command> key is defined as an X “meta” key. |
| -auth *authorization-file* | Specifies a file that contains a collection of authorization records used to authenticate access. See also the *xdm* and *Xsecurity* manual pages. |
| bc | Disables certain kinds of error checking for bug compatibility with previous releases (e.g., to work around bugs in R2 and R3 *xterms* and toolkits). |
| -bs | Disables backing store support on all screens. |
| -rxb *bytes* | Set the receive socket buffer size. Default is 4096. |

| | |
|---|---|
| -cmdkey *alt* | Defines the Macintosh <Command> key as an X “alt” key. By default, the <Command> key is defined as an X “meta” key. |
| -co *filename* | Sets name of RGB color database. |
| -dpi *resolution* | Sets the resolution of the screen in dots-per-inch. To be used when the server cannot determine the screen size from the hardware. |
| -fc *cursorFont* | Sets the default cursor font. |
| -fn *font* | Sets the default font. |
| -fp *fontPath* | Sets the search path for fonts. This path is a comma-separated list of directories that $X^{Ten}$ searches for font databases. It may also indicate a network font server in the form:<br>`tcp/`**`hostname`**`:7000`<br>where *hostname* is the machine name of the network font server. |
| -fs | Starts the server in full screen mode. In this mode, the Macintosh menu bar is hidden by the X desktop. Pull-down menus from the Macintosh menu bar remain accessible by pressing the mouse button while dragging the cursor along the top edge of the desktop. |
| -help | Prints a usage message. |
| -I | Causes all remaining command line arguments to be ignored. |
| -ld *kilobytes* | Sets the data space limit of the server to the specified number of kilobytes. A value of zero makes the data size as large as possible. The default value of -1 leaves the data space limit unchanged. |

| | |
|---|---|
| -ls *kilobytes* | Sets the stack space limit of the server to the specified number of kilobytes. A value of zero makes the stack size as large as possible. The default value of -1 leaves the stack space limit unchanged. |
| -mbm *middle-key* | Changes the middle mouse button keystroke mapping. The key entered as a decimal value representing the Macintosh virtual key code corresponding to the desired key. Allowable values are between 0 and 127. Refer to section "3.2.7 Mouse Button Mapping" for a table of keys and virtual key codes. |
| -mbr *right-key* | Changes the right mouse button keystroke mapping. |
| -menu | Starts the server in "dedicated" full screen mode. In this mode, the Macintosh menu bar is hidden by the X desktop and access to Macintosh pull-down menus is completely disabled.<br>The *xtmenu* program, run from your X window manager menu environment, may be used to temporarily restore access to your Macintosh desktop. Caution: this option locks out access to Finder and other running Macintosh applications! |
| -mo | Specifies that the Macintosh <Option> key will be used in combination with the mouse button keystroke to simulate the middle or right mouse button. When used without the *-mbm* or *-mbr* startup options, the default middle and right mouse button keystrokes become <Option>-left arrow and <Option>-right arrow, respectively. Selecting this option causes all arrow keys, including the up and down arrows, to function normally. |

| | |
|---|---|
| -optkey *alt* or *meta* | Defines the Macintosh <Option> key as either an X "alt" or "meta" key. This options overrides the *-mo* options described above. By default, the <Option> key is undefined as an X key. |
| -su | Disables "save under" support on all screens. |
| -swc | Runs the server with a "software" cursor, allowing X clients to specify a cursor size that exceeds the Macintosh standard 16 x 16 pixel cursor size. |
| -to *seconds* | Sets the default connection timeout in seconds. |
| -wm | Forces the default backing-store of all windows to be "WhenMapped"; an easy way of getting backing-store to apply to all windows. This option can be useful when displaying X applications over slow serial links. |

You can also have X$^{Ten}$ connect to an X display manager using XDMCP, the X Display Manager Control Protocol. Refer to section "6.2.4 X Display Management Under X$^{Ten}$" for more information on the X display manager. The following options control the behavior of XDMCP:

| | |
|---|---|
| -query *host-name* | Enable XDMCP and send "Query" packets to the specified host. |
| -broadcast | Enable XDMCP and broadcast "BroadcastQuery" packets to the network. The first responding display manager will be chosen for the session. |
| -indirect *host-name* | Enable XDMCP and send "IndirectQuery" packets to the specified host. |
| -port *port-num* | Use an alternate port number for XDMCP packets. Must be specified before any *-query*, *-broadcast* or *-indirect* options. |

| | |
|---|---|
| -once | Normally, the server keeps starting XDMCP sessions, one after the other. This option allows the server to continue operating without XDMCP access and control restrictions after a single XDMCP session. |
| -class *display-class* | XDMCP has an additional display qualifier used in resource lookup for display-specific options. This option sets that value, by default it is "MITUnspecified". |
| -cookie *xdm-auth-bits* | When testing XDM-AUTHENTICATION-1, a private key is shared between the server and the manager. This option sets the value of that private data. |
| -displayID *display-id* | An XDMCP specific value allowing the display manager to identify each display so that it can locate the shared key. |

# APPENDIX E
## X$^{Ten}$ Security & Passwords

X$^{Ten}$ is factory-configured with two user accounts — *XTen* and *root*. The password for both accounts is “`XTen`”.

When X$^{Ten}$ is run in the default configuration mode it uses the *XTen* account and no password is required, so typical users will not need to change and memorize an X$^{Ten}$ password. However, for environments that require more security, X$^{Ten}$ may be optionally configured to require a password during startup (see section “6.2 X Server Startup” for more information). Also, X$^{Ten}$ may be optionally configured to permit remote login, perhaps to enable access for a network administrator (see section “2.4.2.1 Enable Incoming Connections” for more information). In either of these cases, the default passwords should be changed to prevent unauthorized access to the X$^{Ten}$ system by these well-known default passwords.

### Changing the XTen Password

To change the password for the default *XTen* account, start X$^{Ten}$ and start a local *xterm* (use <Command>-<Option>-T). Use the *passwd* command and enter the old password followed by the new password. The *passwd* command does not display the passwords as they are being entered, and thus requires entering the new password twice to ensure it was entered properly.

```
XTen@ppc0% passwd
Changing local password for XTen.
Old password:XTen
New password:<Your Password>
Retype new password:<Your Password>
passwd: rebuilding the database...
passwd: done
```

## Changing the root Password

To change the password for the superuser or *root* account, start X$^{Ten}$ and start a local *xterm* (use <Command>-<Option>-T). Use the *su* command to become the superuser. The *su* command requires entry of the root password "XTen", and does not display the password as it is being entered. Once you have become the superuser (note the change in the prompt string) use the *passwd root* command and enter the new password. The *passwd* command does not display the passwords as they are being entered, and thus requires entering the new password twice to ensure it was entered properly. After changing the superuser password, exit from the *su* command to return to the normal *XTen* user.

```
XTen@ppc0% su
Password:XTen
XTen@ppc0# passwd root
Changing local password for root.
New password:<Your superuser password>
Retype new password:<Your superuser password>
passwd: rebuilding the database...
passwd: done
XTen@ppc0# exit
XTen@ppc0%
```

# APPENDIX F

## References, Acknowledgments & Suggested Reading

- OSF/Motif User’s Guide, Open Software Foundation

The following books published by O’Reilly & Associates are recommended:

- Volume 3: X Window System User’s Guide
  Details the X Window System and common X client applications.
  This volume is available in an OSF/Motif Edition focusing on the Motif user environment and *mwm*.

- Volume 8: X Window System Administrator’s Guide
  Comprehensive guide to administering the X Window System. This volume is a highly recommended companion to the X$^{Ten}$ software package.

The following Nutshell Handbooks from O’Reilly & Associates my also be helpful:

- The X Window System in a Nutshell
- The Whole Internet User’s Guide
- Essential System Administration
- TCP/IP Network Administration

# APPENDIX G

## Using OpenTransport or Replacing OpenTransport

X$^{Ten}$ has been programmed with two alternative TCP/IP protocol implementations. The first is an implementation of the standard sockets interface which has been mated with MacOS OpenTransport. In this configuration, calls to the sockets interface are translated into one or more calls onto the MacOS OpenTransport TCP/IP implementation. In addition, the results of any OpenTransport calls are translated into sockets operations. This is the default configuration. It provides for complete compatibility between X$^{Ten}$ networking applications and other MacOS networking applications. This configuration allows X$^{Ten}$ to use OpenTransport PPP, creating an easy setup for using X Windows over modems and telephone lines.

The second TCP/IP configuration provides for the replacement of OpenTransport with Tenon's TCP/IP stack. This implementation is an industrial-strength BSD UNIX TCP/IP implementation containing many features not found in OpenTransport, such as multi-link, multihoming.[1]

A third alternative is to simultaneously use both OpenTransport and Tenon's TCP/IP stack. The advantages of this configuration are that both Macintosh applications and X$^{Ten}$ can operate using their native TCP implementations. The disadvantage is that two IP addresses are required, one for OpenTransport and a different one for Tenon TCP. A side effect of this configuration is that the two TCP stacks cannot talk to each other, but this is typically not a requirement for X$^{Ten}$ systems.

The Tenon TCP stack is more robust, better-performing, and more fully-featured than Apple's OpenTransport. When Tenon TCP is running, Macintosh applications that are accustomed to using MacTCP will automatically use the TCP protocols in Tenon TCP.

The choice of using OpenTransport, Tenon TCP or dual stack networking is controlled by the placement of three libraries in the *Tenon Kernel Libraries* folder. The following table shows the valid combinations for the location of these libraries. To change configuration, quit X$^{Ten}$, use the Finder to re-position the libraries, and re-launch X$^{Ten}$.

[1] OpenTransport 1.3 includes single-link, multihoming.

### Networking Configuration Choices

| Configuration | Tenon Kernel Libraries | Disabled |
|---|---|---|
| Use OpenTransport (default) | OT SocketLib | SocketLib<br>MacTCPdLib |
| Replace OpenTransport | SocketLib<br>MacTCPdLib | OTSocketLib |
| Dual stack (requires two IP addresses) | SocketLib | OTSocketLib<br>MacTCPdLib |

# INDEX