```
rsh -n $HOST hdslocal hdsoclock '&'
rsh -n $HOST hdslocal hdsmemory -iconic '&'
fi
#
# create LOGIN window
xterm -title LOGIN -geometry 80x24+10+10
....
```

These examples can be customized further as your system needs require. The principal advantage of Xdm is its management of the user's session, so the system administrator can automatically provide a complete session environment for each user.

```
setenv LD_LIBRARY_PATH  $OPENWINHOME/lib
setenv X11ONLY=YES
setenv MANPATH  $OPENWINHOME/share/man:/usr/man
set path=( $OPENWINHOME/bin  $OPENWINHOME/bin/xview  $path )
echo "Starting OPEN LOOK Window Manager"
echo "OLWM DISPLAY = $DISPLAY"
$OPENWINHOME/bin/olwm &
```

This example uses simple settings and "default" paths and directories; your installation may use different locations and directories. This shell script must be executable and should be in your home directory. Your entry in the .xsession file would be **do.olwm** instead of the **twm&** command in the previous example.

3.3.6.1 Vendor Display Class String

If you are using Indirect or Broadcast xdm requests, you may need to enter a vendor display class string for the ViewStation in the DisplayManager.servers resource. The HDS ViewStation display class string is "HDS-ViewStationFX". The manufacturer’s class string is "-Ethernet-%s" where %s is your ViewStation’s Ethernet hardware address in hexadecimal form, *nn:nn:nn:nn:nn:nn*.

3.3.6.2 ViewStation .xsession File

The xsession file shown above is a very simple implementation of the Xdm startup file. In practice, your xsession file would probably contain some code to analyze the display type (for color or monochrome differences), or for display capability (HDS ViewStation displays with local clients and other devices without this capability). This could be done by node naming distinctions or by analyzing the device’s vendor string.

```
....
# extract a host name from $DISPLAY
HOST=‘echo $DISPLAY | sed “s/”.*$//”‘
# check for HDS ViewStations
# all HDS ViewStations are color units
# all HDS ViewStations use local clients
if expr $DISPLAY : “hds*”
then
xsetroot -solid “PaleGreen”
xrdb -merge $HOME/.Xdefaults-hds
xmodmap -e keymap-hds
rsh -n $HOST hdslocal mwm ‘&’
```

The user customizes his display environment by using an individual .xsession file. This file contains a list of user commands or selections appropriate for the user's environment, such as window manager selection, clients to be run, etc.

The .xsession file is located in the user's home directory and is run when xdm starts up. (There is a general .Xsession file created by the system administrator that contains sufficient information to run the system and your display with default values without the customization of an individual .xsession file.)

A typical .xsession file might contain some entries like these:

```
xrdb -merge .Xdefaults
xmodmap -e “keycode 102 = Delete”
xclock -fg red -bg white -geom -0+0 &
xterm -e "rsh hdssun5" -fg orange -bg black &
xterm -fn 10x20 -bg blue -fg white &
twm&
xterm -title LOGIN -geometry 80x24+10+10
```

You should notice several things about this file. The first line is the command to load the resource file. The next lines start some clients, an xclock, an xterm on a remote host, an xterm with a big font, and a window manager. (Additional levels of customization can occur here, since each of the clients has an app-defaults file (found in */usr/lib/X11/app-defaults*) which can be used for individual settings, such as scroll bars for the xterms, or special emulations, etc. These customizations are often most important for the window manager's operation, which often uses a run command file, such as .twmrc, found in the user's home directory.)

The most important thing to notice about the .xsession file is that the last entry cannot run in the background. If you assign this last client to be a background process, xdm will detect the process as not active and assume you are finished and will therefore close your session. In this example, the last process is an xterm window with the label LOGIN in the title bar. This means that you can open or close any number of clients and windows in your session, but as soon as you close the LOGIN window, xdm will close your session. You could use your window manager or a client like xclock as your LOGIN window, but the same rules apply; when that window closes, the entire session is over.

You may also use the xsession file to execute shell script files, which might be used to set up and start a window manager; for instance, your shell script file might contain the various environment variables and path settings needed by the OPEN LOOK or Motif window managers. The xsession file itself must be executable, which you can do with the chmod command (**chmod 777 .xsession**).

For instance, you could create a simple shell script, called do.olwm, to start the OPEN LOOK window manager. This script might contain some environment settings and the command to start the window manager:

```
setenv OPENWINHOME  /usr/openwin
```

The **Retry Action** field shows whether you want the ViewStation to keep trying to get the xdm display manager to respond or not. The values are Persist and Stop. If you want to use xdm, you probably want the requests to persist until you get a response.

The **Minimum KeepAlive time** field gives the minimum time the ViewStation waits after it has received its last message (of any kind) from the host computer before sending a KeepAlive packet to the display manager. This lets the ViewStation continue its session even if there is no activity from the host computer. The value is an integer value in minutes, with a default value of one minute. This means that the ViewStation sends a KeepAlive packet to the display manager after 1 minute of inactivity. It sends another after 2 minutes, another after 3 minutes, and so on. Any ViewStation activity from the keyboard or the mouse will wake up the session again and reset the KeepAlive time to 1 minute. The net effect is that your session stays alive even if you leave for the weekend.

The **KeepAlive retry timeout** field sets the time the ViewStation waits for a response from the host after it has sent its KeepAlive packet. This is an integer value in seconds, with a default value of 20 seconds. The ViewStation waits 20 seconds after its request for a response; if it doesn't receive one, it closes the connection because the display manager hasn't answered. After it sends the first request (at one minute of inactivity) it sends another request after 2 seconds, another after 4 seconds, and so on until the retry timeout value is reached.

The **Default Access Control** field sets the ViewStation's Access Control security, also called MIT Magic Cookie. The default setting is Enabled. Access control means that Xdm establishes an .Xauthority file with the initial logon. This file is used for security control of other login connections. Refer to Section 1.11.3 for more details and methods of using or modifying Access Control security.

The **Reset Server at End of Session** field selects whether the ViewStation's server software resets itself at the end of an Xdm session. The default setting is Enabled. Depending on your system configuration, this may or may not be desireable, since a server reset removes a lot of the user's settings.

These settings and timeout values should be worked out with consideration of the settings for the corresponding values for xdm on the host computer.

Note also that all of these xdm variables are saved in NVM because they are used very early in the power-up process. If you change the IP Broadcast address value, which can be changed by Apply Changes for the current session only, xdm will use the IP Broadcast address set in NVM when it starts. If you are using the Broadcast address for xdm, you should save its value in NVM each time you edit the field.

3.3.6 Using the .xsession File

There are two Xsession files, an Xsession file in the Xdm directory that is a system-wide default file and an .xsession file that is in the user's home directory. Usually the .xsession is the only file of interest to the user. The Xsession file is controlled by the system administrator and is used only if the .xsession file is not found.

Direct specifies that you want to communicate only with the xdm client running on the host shown in the XDMCP Server Address field.

Indirect specifies that you want the host given by the XDMCP Server Address field to act as an Indirect host. This Indirect host usually forwards the request to another host computer. You can also use the local HDS Chooser client to give you a menu of available hosts.

Hosts access method specifies a file of connection options in a host file. The "Hosts" access method allows you to specify multiple hosts with individual access methods for your XDMCP request. You can set up an Xdm Hosts file by selecting the Hosts access method and entering the path to your Xdm hosts file in the "Hosts File" line. A sample file is given below:

The Xdm hosts file contains a list of hosts, one per line. Each host entry can be one of three XDMCP query types - Direct, Indirect, or Broadcast. Host names starting with an asterisk ('*') are queried via Indirect requests. Host names not starting with an asterisk are queried via Direct requests. A host name of 'BROADCAST' tells the ViewStation to perform a broadcast request. Each host entry also has an associated retry timeout. The timeout parameter specifies how many seconds to send requests to a host before trying the next host in the list. The default timeout is 0 seconds. Hosts are tried in the order in which they appear in the hosts file.

Characters after a '#' character are comments.

This is a sample XDMCP hosts file:

```
# XDMCP hosts example file

*hdssun1 # Indirect request to hdssun1, timeout=0

hds486 timeout=10 # Direct request to hds486

hdshp timeout=15 # Direct request to hdshp

BROADCAST # Broadcast request
```

This file specifies that at start_time + 0 seconds, an indirect request is sent to hdssun1 and a direct request is sent to hds486. At start_time + 10 seconds, a direct request is sent to hdshp in addition to requests to hdssun1 and hds486. At start_time + 25 seconds, a broadcast request will be sent in addition to requests to hdssun1, hds486, and hdshp.

The **Run Local Chooser** field lets you select whether you run the HDS ViewStation's local Chooser client. When you use the Indirect or Broadcast access for Xdm, this client puts up a menu of available Xdm hosts. It tests the network to see which hosts respond to your Xdm request and then lets you select the one you want. This is a client running on the ViewStation. It is described in detail in Section 2.6.4. It works with both R4 and R5 Xdm versions.

The **XDMCP Server Address** field shows the IP address of the Direct or Indirect Access host shown in the previous field. The default value is 0.0.0.0.

You can start the ViewStation's Chooser client from the XDMCP menu in Setup Mode. Its operation is described fully in Section 2.6.4.

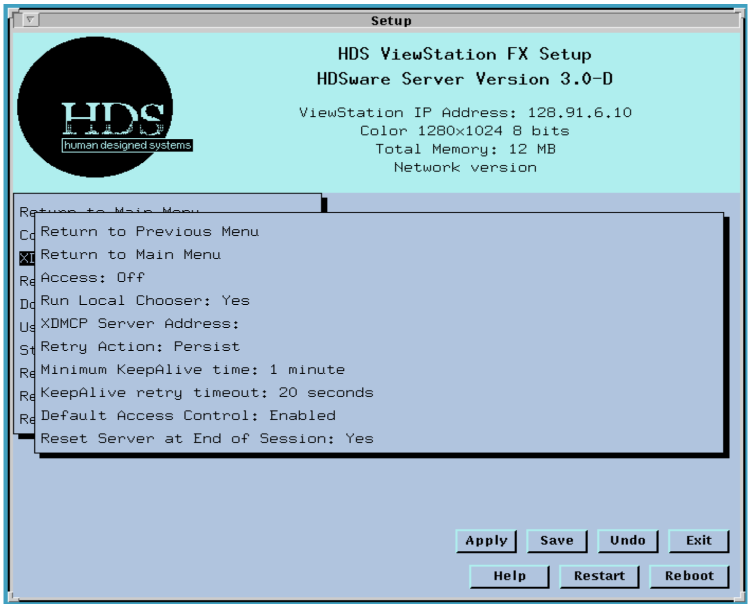
3.3.5 Using XDM on the ViewStation

The ViewStation will use Xdm if you select Xdm access on the XDMCP Options menu of Setup Mode. With an access method selected, the ViewStation sends an Xdm request each time it starts (or the server resets). Response to this request by the Xdm process begins Xdm's control the ViewStation's display. Note that any processes running on the ViewStation (local clients, including Setup Mode) will be killed by Xdm and won't be available until the Xdm login is complete.

3.3.5.1 XDMCP Options Selection in Setup Mode

XDMCP (X Display Manager Control Protocol) is a host-based handshaking protocol to establish a connection with a host and to manage your display. It works with a host client to control your logon process, start the clients you specify, and set up configuration parameters whenever you turn on your ViewStation.

The XDMCP Options Menu is shown below:



HDS ViewStation FX Setup
HDSware Server Version 3.0-D
ViewStation IP Address: 128.91.6.10
Color 1280x1024 8 bits
Total Memory: 12 MB
Network version

Return to Previous Menu
Return to Main Menu
Access: Off
Run Local Chooser: Yes
XDMCP Server Address:
Retry Action: Persist
Minimum KeepAlive time: 1 minute
KeepAlive retry timeout: 20 seconds
Default Access Control: Enabled
Reset Server at End of Session: Yes

The **Access Type** field specifies whether you use xdmcp and which host you want to have manage your display. There are five values for this field:

Off (default value) specifies that you do not want to use XDMCP.

Broadcast specifies that you want to accept an offer to manage your display from the first host that answers your broadcast request (your IP Broadcast Address is set in the Communications Options Menu). You can use the local HDS Chooser client to give you a menu of available hosts.

If you have problems with your Xdm process, it may be helpful to look at its version and operation. Look at the Xdm man pages on your system.

In summary:

X11R3 selected hosts to manage by looking at the Xservers file. This management was set by the system administrator and had no user selectivity.

X11R4 changed the functionality to permit the user's selection of Xdm management and the method of making the connection. This was called XDMCP.

X11R5 improved Xdm control of the management process and added an Xdm "Chooser" client to aid in selecting an Xdm host.

3.3.4 Using Xdm with your ViewStation

Using Xdm with the ViewStation is simple. There are only two requirements; first, that Xdm be running on your host, and second, that you request Xdm access from the XDMCP menu of Setup Mode.

There are several methods of Xdm access.

3.3.4.1 Xdm Direct Access

In this case you select the Xdm Access as Direct and specify a host address for the Xdm host. The ViewStation sends its Xdm request directly to that host and an Xdm session is established.

3.3.4.2 Xdm Indirect Access

Indirect access for Xdm means that the ViewStation can connect to one of several host computers, with one computer serving as a redirection agent for the system. Typically a chooser box appears with a fixed list of Xdm hosts. The user's selection is then forwarded to that host.

3.3.4.3 Xdm Broadcast Access

When Broadcast is selected as the access method, the ViewStation broadcasts an Xdm request. All Xdm hosts that respond to the broadcast are put into the Chooser box and the user can make the selection of the Xdm host.

3.3.4.4 Xdm Chooser

Xdm, with version X11R5, has a Chooser client. This is a selection box that lists the Xdm hosts that are available, generally from an Indirect or Broadcast Xdm access selection.

3.3.4.5 ViewStation's Chooser Client

The ViewStation has a local Chooser client. The ViewStation's Chooser works the same way as the host-based chooser, but it has two important differences. The ViewStation's chooser works with both X11R4 and X11R5 Xdm processes and it allows the user to enter an Xdm host into its selection box. These are often very convenient features.

```
:0 Local local /usr/bin/X11/X :0
```

The Xservers file was the way Xdm managed network devices under X11R3. With X11R4 and R5, individual devices send an Xdm request.

3.3.2.4 Xsetup File

The Xsetup file is a new file with X11R5. In most cases, it is empty. It allows the user to specify any startup commands that should be run BEFORE the Xdm banner is up and the user logs in. This would typically be something like xmodmap that would put special keyboard modifications in place, such as foreign-language mappings, that the user would need to type in his name and password.

The ViewStation has a corresponding file, the server-reset file, that allows this kind of configuration. The ViewStation also has a local xmodmap client, refer to Section 2.6.13, which can be used for this purpose.

3.3.2.5 Xstartup File

The Xstartup file contains startup commands that should be executed AFTER the user has logged in and is verified. This file is typically empty.

3.3.2.6 Xreset File

The Xreset file contains commands that should be executed after the user’s login has been verified.

3.3.2.7 xdm-errors File

The xdm-errors file contains error messages relative to Xdm operation. If you have problems with Xdm, look in this file.

If you have more extensive Xdm problems, restart Xdm with the -debug argument set (with a fairly high value, say 30). This will give extensive error reporting for the Xdm process.

3.3.2.8 xdm-pid

The xdm-pid file stores the process ID for the parent Xdm process. This is sometimes an important value, since each Xdm connection spawns a daughter process and it is difficult to determine which is the parent process.

There is a resource in the Xresource file which permits changing this file’s name.

3.3.2.9 Xsession file

The Xsession file (as distinct from the .xsession file) is the system’s default startup script. If there is no user’s .xsession file, this .Xsession file will be used.

In the standard MIT release, this file starts an xterm and the twm window manager. Of course, your system administrator can modify this script.

3.3.3 Xdm Versions

There are a number of different versions of Xdm. These are substantially different depending on their chronology, whether X11R3, R4, or R5. There are also some differences in vendor releases, though these differences are generally minor.

3.3.2 Xdm Configuration Files

These are the configuration files for X11R5 XDMCP. Older Xdm systems won’t have all of these files.

The xdm-config file contains resources specifying xdm related data files;

```
DisplayManager.servers:          /usr/lib/X11/xdm/Xservers
DisplayManager.errorLogFile:     /usr/lib/X11/xdm/xdm_errors
DisplayManager*resources:        /usr/lib/X11/xdm/Xresources
DisplayManager*setup:            /usr/lib/X11/xdm/Xsetup
DisplayManager*startup:          /usr/lib/X11/xdm/Xstartup
DisplayManager*reset:            /usr/lib/X11/xdm/Xreset
DisplayManager*session:          /usr/lib/X11/xdm/Xsession
DisplayManager*authorize:        false
```

3.3.2.1 Xdm Authorize

The *authorize* resource controls the user-based access control (different from host-based access control governed by *xhost*). The example in the paragraph above has access control turned off.

The following resources set access control on for the local display and then using the unqualified authorize afterward to set host-based access control for other displays.

```
...
DisplayManager._0.authorize: true
DisplayManager*authorize: false
...
```

Access control is discussed from the ViewStation’s point of view in Section 1.11.3.

3.3.2.2 Xresources File

This file contains resources used by Xdm, typically for its login banner. If you want to change these resources, for instance by adding your personalized greeting, you do it by modifying entries in this file.

Look in this file to see the resource names and settings.

3.3.2.3 Xservers File

The Xservers file contains a list of displays which Xdm must manage. These are typically displays which do not have XDMCP and thus cannot select to have Xdm manage them. Normally the ViewStation selects whether Xdm manages its display or not, based on the user’s selection in the XDMCP menu in Setup Mode.

This file is usually empty except for a single entry for the host computer’s local display:

XDMCP Operations

This section describes the operation of XDMCP, the X Display Manager Command Protocol, which runs oun your host computer and manages the ViewStation's display.

Section 3.0 - Network and Host System Operations

3.3 XDMCP Operations

Xdm is a host-based program that manages the X terminal's display and session. It has several functions:

- Presents a login screen; verifies user login and password
- Starts the user session after verification
- Sets up the X environment
- Terminates the environment properly and puts up the login screen again

There are some variants of Xdm that have substantial differences in operation from Xdm but still provide the overall session management. Some of these are the VUE environment by Hewlett Packard, DXsession by Digital Equipment Corp., and Toolchest by Silicon Graphics. These are similar in function to Xdm, but some of the files and syntax may be very different. Consult your man pages for operation of these systems. The ViewStation's "manage my display" request is recognized by all of these systems.

3.3.1 Starting Xdm on your Host

You must be the root user to start Xdm. The Xdm executable and configuration files are in */usr/lib/X11/xdm*. Look at the configuration files to be sure they reflect your system environment. Actual startup will probably be from an *rc* or *rc.local* file on your host.