

Chapter 1

Configuring XRemote

1

This chapter focuses on Cisco's support for the XRemote capabilities of Network Computing Devices, Inc. (NCD) X Window terminals. It covers the following information:

- Outline of the X Protocol and the basics of the XRemote capability
- Basic XRemote automated and manual start-up procedures
- Terminal server configuration for XRemote
- A summary of XRemote global configuration commands

This chapter does not fully address the details of the X Window System and X-based computing environments. The following are suggested sources for more complete definitions and implementation descriptions:

- *X Library Reference Guide*, Ira Chayut and Camile Cook. A System Publications, Inc.
- *X Tool Kit Reference Manual*, Ira Chayut and Camile Cook. A System Publications, Inc.
- *X User Reference Guide*, Ira Chayut and Camile Cook. A System Publications, Inc.
- *X Window System User's Guide for Version 11*, Tim O'Reilly, Valerie Quercia, and Linda Lamb. O'Reilly and Associates, Inc.

The X Window System

The X Window System™, typically called X, is a network-based graphics window system developed at the Massachusetts Institute of Technology (MIT) in 1984 for workstations running UNIX. It was originally written in the C programming language; however, X is not tied to a specific operating environment, programming language, or hardware configuration.

Previous window systems were *kernel-based* and were thus closely linked to the operating system running on a workstation itself. They typically have run only on discrete systems, such as a single workstation. The X Window System is not part of any operating system, but instead is composed of application programs. Thus, X enables flexible, graphics-based network computing across a wide range of operating systems and hardware platforms.

X and the Client Server Model

The underlying architecture of the X Window System is based on a *client server* model. The system is split into two parts: *display servers* and *clients*. Display servers provide specific display capabilities and track user input; clients are application programs that perform specific tasks. These two parts can reside on the same computer, or can be separated over a network. In an X terminal environment, such as in NCD terminal implementations, the display server resides on the display station and the client resides on a host computer.

Because X employs this functional partitioning and is independent of both hardware and operating environment, X terminal users can access different types of computers to simultaneously access several applications and resources in a multivendor environment. A user at an X terminal can run and display a calendar program on a VAX, a spreadsheet program on a PC, and a compiler on workstation concurrently.

How XRemote Works

XRemote is a protocol developed specifically to optimize support for X over a serial communications link. Its compression and decompression algorithms are designed to handle bit-mapped displays and windowing systems.

There are two basic parts to XRemote:

- Local PROM-based elements
- The XRemote helper process

The PROM-based elements consist of the *X server* and *boot monitor*. These are responsible for booting the display station, managing local processes, and handling communications with the host computer. The helper process allows the XRemote display station to communicate, through the terminal server, with other hosts on the network. PROMs that implement these elements are available from NCD.

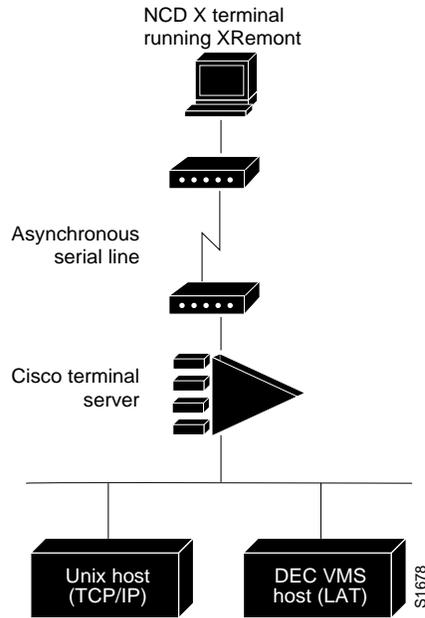
Cisco's Implementation of XRemote

XRemote enables a user of an NCD network display station to run the X Window System via 9600 baud (and faster) modem connections with superior performance compared with conventional serial protocols, such as SLIP (Serial Line Internet Protocol). The NCD terminal must have a compatible XRemote PROM set installed.

Figure 1-1 illustrates XRemote network accessibility via Cisco terminal servers.

Cisco's implementation of XRemote is fully compatible with the NCD XRemote protocol. Remote access to fonts is provided using the Internet standard Trivial File Transfer Protocol (TFTP).

Figure 1-1 Example XRemote Connection Illustration



Starting XRemote

XRemote can be invoked in one of two ways:

- Automatically using the XDMCP (X Display Manager Control Protocol)
- Manually via a step-by-step access process

The following brief procedures outline steps required for starting up XRemote in several typical environments. When possible, it's generally suggested that the automated XDMCP process be used.

Automatic Session Startup

If your host computer includes a server for XDMCP, such as the *xdm* program included in X11R4, you can use automatic session startup to start up your XRemote session.

The EXEC command used to initiate XRemote using XDMCP is **xremote xdm**. This command has the following syntax:

xremote xdm *hostname*

This command causes an XDMCP session start-up request to be made to the computer specified (*hostname*). If a host name is not specified, a broadcast message is sent to all hosts. The first host to respond by starting up a session is used.

The terminal server and X terminal stay in XRemote mode until either the display manager terminates the session, or a reset request is received from the X terminal.

Manual Session Startup

If you are not using a host computer which supports XDMCP, you must use manual session startup. Manual session startup involves several steps:

- Step 1:** Enabling XRemote manually on the terminal server port
- Step 2:** Connecting to the host computer
- Step 3:** Setting the location of the X display
- Step 4:** Starting up client application(s)
- Step 5:** Returning to the terminal server EXEC prompt
- Step 6:** Enabling XRemote manually again on the terminal server port

Enabling XRemote Manually

The EXEC command **xremote** prepares the terminal server for your use of Xremote. The terminal server replies with a message informing you of your X display location:

```
Xremote enabled; your display is darkstar:2018
Start your clients and type Xremote again
```

In the above message, the terminal server informs you that your display is `darkstar:2018`. This information will be used in a following step.

Connecting to the Host Computer

Connect to the host computer using the **telnet**, **lat**, or **rlogin** commands, and log in normally.

Setting the Location of the X Display

Note: If you are using a version of Telnet (on the remote host) which supports the “X Display Location” option (RFC 1096), you may skip this step; proceed to “Starting Clients.”

The next step is to inform the host computer of the location of your X display, which was provided to you by the terminal server when you entered the `xremote` command above.

For most versions of the UNIX operating system, the X display location is set by using the `setenv` command to set the environment variable `DISPLAY`. Refer to the on-line, X(1) manual page for more information.

Example:

```
host_prompt% setenv DISPLAY darkstar:2018
```

On VAX/VMS, use the SET DISPLAY command. Refer to the *VMS DCL Dictionary* for more information.

Example:

```
$ SET DISPLAY/CREATE/NODE=DARKSTAR/SERVER=2018/TRANSPORT=[transport]
```

Note: You must have installed either the TCP/IP transport from DEC, or a third-party TCP/IP transport. Contact your VAX/VMS system administrator for the appropriate LAT or TCP/IP transport name. If you use the LAT transport, use the terminal line number (22) instead of the display number 2018.

Starting Clients

At this point, you must start your client applications for your host operating system.

The terminal server accepts the X connection attempt from the client application, and places the client into a dormant state.

Returning to the Terminal Server EXEC Prompt

If it is possible to log off of the host computer and keep your X clients running in the background, you can do so now. This conserves resources on the host and terminal server that would otherwise be inaccessible until you exited from XRemote state.

If not, you may simply escape back to the terminal server prompt using the escape character (Ctrl ^, Shift-x).

Re-enabling XRemote Manually

Enter the command **xremote** to access XRemote mode. If the X client(s) connected successfully, the session will be put into XRemote mode, and the clients will be allowed to complete their startup.

If no clients were found, you will see the following message:

```
No X clients waiting - check that your display is darkstar:2018
```

Check your hosts to determine if an error was made when starting the session. The most likely cause is an improperly specified display location. Another possible cause is the host computer does not recognize the name of the terminal server through which you are attempting to make a connection.

Session Termination

In manual operation, the terminal server and X terminal remains in XRemote mode until either all clients disconnect or a reset request is received from the X terminal.

If a session terminates during startup, it may be because you invoked transient X clients which set some parameters and disconnected. There must always be one session connected or the connection will be reset.

Configuring XRemote

To make a host connection via NCD's XRemote feature and a Cisco terminal server, you must complete some basic configuration steps.

- Step 1:** Configure the X-terminal and terminal server for fonts to be used and other special parameters.
- Step 2:** Cable and set up your modem for use with XRemote (access over asynchronous lines only), or cable the NCD terminal directly to the Cisco terminal server.
- Step 3:** Set up the NCD terminal's X Server, Network, and Serial Parameters. Refer to your NCD manual for specific details of configuring these parameters.

Font Selection

Depending on the fonts to be used on the terminal, you may need to specify the location of a font server. The NCD terminal contains a small set of built-in fonts in local ROM. It is recommended that these fonts be used, as loading fonts over a serial line can increase application start-up time. The default for an NCD terminal is to use built-in fonts.

Accessing Non-Resident Fonts

When an X terminal application requests a font which is not stored in the NCD ROM, the X terminal makes a request for a font file from the terminal server. The terminal server uses the trivial file transfer protocol (TFTP) to load the font from the font server, and then passes the font to the X terminal using the XRemote protocol. The process of loading fonts from the terminal server to the X terminal can take 30 to 45 seconds, depending on the size of the font file.

An X server can display only the fonts it finds in the directories in its font path. The X server's default font path includes only the built-in fonts. To access fonts stored on a host, you must add the host's font directories to the X server's font path. To do this, use the UNIX command `xset` with the `[fp+]` argument to add fonts to the end of the server's font path.

For example, to allow your display station to access the 100 dpi (dots per inch) fonts found in the standard font directory, run the following command (at the host system prompt):

```
host_prompt%xset fp+ /usr/lib/x11/ncd/fonts/100dpi
```

For more information, refer to the *NCDware XRemote User's Manual*.

Terminal Server Configuration

Using three global configuration commands, several terminal server options can be set that identify font loaders (hosts), set buffersize, and specify the number of font loader retries. These system configuration commands are available only when the terminal server privilege level has been enabled, allowing access to the **configure** command. Once you've accessed the system configuration facility, use the **xremote tftp** global configuration commands to modify these parameters. The only parameter that generally must be set is the font loader host identification (only if you intend to download fonts to the X terminal).

Identifying Font Servers

Use the global configuration command **xremote tftp host** to add a specific font server as a source of fonts for the terminal (and the terminal server). This command has the following syntax:

```
xremote tftp host [hostname]  
no xremote tftp host [hostname]
```

Each time a new host name is entered, the list on the terminal server is updated. Font servers are queried in the order of their definition when the X terminal requests a font.

The command **no xremote tftp host** [hostname] removes a font server from the list maintained by the terminal server.

Specifying Buffersize

Use the global configuration command **xremote tftp buffersize** to set the terminal server buffer size for specific font requirements. This command has the following syntax.

```
xremote tftp buffersize [decimal value]  
no xremote tftp buffersize
```

This **xremote** command sets the buffer size used for loading font files to a specified value (in decimal bytes). When the X terminal requests that a font file be loaded, the terminal server must first load the font file into an internal buffer before passing it to the X terminal. The default value of 32768 is adequate for most font files, but the size can be increased as necessary for non-standard font files.

The buffer size may be set as low as 4096 bytes, and as large as the available memory on the terminal server will allow.

The command **no xremote tftp buffersize** restores the buffer size to the default of 32768 bytes.

Determining Font Loader Retries

Use the global configuration command **xremote tftp retries** to specify the number of retries the font loader will attempt before declaring an error condition. This command has the following syntax:

```
xremote tftp retries [value]  
no xremote tftp retries
```

The default value is 3. Under certain conditions, you may need to increase this number, particularly if the font server(s) is(are) known to be heavily loaded. Acceptable values range from 1 to 15.

The command **no xremote tftp retries** restores the retries number to the default of 3.

Making Modem Connections

The following brief discussion outlines recommendations for modem setup when using XRemote to access remote resources via a Cisco terminal server.

Modem Requirements

In general, you can use any modem that provides acceptable performance for your application. The following guidelines apply to XRemote operation using a modem:

- Disable any error correction and compression features of the modem. As XRemote implements its own compression and error correction, the modem's compression and error correction actually impair performance.
- If you must use a flow control mechanism, hardware flow control (such as RTS/CTS or DTR/DSR) is recommended. Software flow control (such as XON/XOFF) is discouraged.
- The modem should incur minimal delays in round-trip transmissions, even when transmitting small packets, and should be transparent to the data stream.
- The modem should provide true full-duplex transmission at 9600 baud or faster. Half-duplex modems are not suitable for use with XRemote.

Monitoring XRemote Connections

You can use the EXEC command **show xremote** to list XRemote connections and monitor XRemote traffic through the terminal server. The syntax for this command is as follows:

```
show xremote
```

This **show** command provides XRemote parameters applied to the entire system, as well as statistics that are pulled for all active XRemote connections (listed by terminal server line). The following example illustrates screen output resulting from a **show xremote** command when XRemote is enabled on a terminal server, and XRemote sessions are active.

Example:

```
ts_prompt>show xremote
Xremote server-wide parameters:
  Font buffersize:      72000           Font retries: 3
  Font memory errors:  0

Font load statistics for host 131.108.1.111:
  Bytes read:          2697239         Files read: 258
  Network errors:      4               File errors: 0

Xremote statistics for tty2:
  Current clients:     9               Total clients: 17
  Requesting client:   5               Current request size: 0
  Replying client:     6               Current reply size: 0
  XDM state:           10              Next timeout: 172460
  Retransmit counter:  0               Local UDP port: 53616
  Keepalive dormancy: 180              Session id: 94
  Unread input:        0               Unwritten output: 0
  Input buffer size:   1024             Output buffer size: 108
  Protocol version:    2               Line state: Connected
  Transmit packets:    50768            Receive packets: 49444
  Transmit errors:     0               Receive errors: 37
  Retransmissions:    403              Receive out of sequence: 76
  Round trip time:     383              Retransmit interval: 766
  Transmit window:     7               Receive window: 7
  Transmit next:       6               Receive next: 3
  Transmit unacked:    6               Receive unacked: 0

Connection 0 - TCP connection from 131.108.1.55 [Display Manager]
  Client state:        CS_ACTIVE        Byte order: MSBfirst
  Unread input:        0                Unwritten output: 0
  Input buffer size:   1024             Output buffer size: 1024

Connection 1 - TCP connection from 131.108.1.55
  Client state:        CS_ACTIVE        Byte order: MSBfirst
  Unread input:        0                Unwritten output: 0
  Input buffer size:   1024             Output buffer size: 1024

Connection 2 - TCP connection from 131.108.1.55
  Client state:        CS_ACTIVE        Byte order: MSBfirst
  Unread input:        0                Unwritten output: 0
  Input buffer size:   1024             Output buffer size: 1024
```

You can use the EXEC command **show xremote line** to list XRemote connections and monitor XRemote traffic for specific lines on a terminal server. The syntax for this command is as follows:

show xremote line *number*

The variable *number* is a decimal value representing the physical asynch terminal lines available on a terminal server. The following example illustrates screen output resulting from a **show xremote line** command when XRemote is enabled on a terminal server, and XRemote sessions are active. Only information specific to an individual terminal line is provided.

Example:

```
chaos#show xremote line 3
Xremote statistics for tty3:
  Current clients:      11          Total clients: 19
  Requesting client:   10          Current request size: 0
  Replying client:     10          Current reply size: 0
  XDM state:           10          Next timeout: 173304
  Retransmit counter:  0           Local UDP port: 28384
  Keepalive dormancy: 180         Session id: 29
  Unread input:        0           Unwritten output: 0
  Input buffer size:   1024        Output buffer size: 108
  Protocol version:    2           Line state: Connected
  Transmit packets:    28875       Receive packets: 18644
  Transmit errors:     0           Receive errors: 13
  Retransmissions:    53           Receive out of sequence: 41
  Round trip time:     384         Retransmit interval: 768
  Transmit window:     7           Receive window: 7
  Transmit next:       2           Receive next: 7
  Transmit unacked:   2           Receive unacked: 0

Connection 0 - TCP connection from 131.108.1.27 [Display Manager]
  Client state:        CS_ACTIVE    Byte order: MSBfirst
  Unread input:        0           Unwritten output: 0
  Input buffer size:   1024        Output buffer size: 1024

Connection 1 - TCP connection from 131.108.1.27
  Client state:        CS_ACTIVE    Byte order: MSBfirst
  Unread input:        0           Unwritten output: 0
  Input buffer size:   1024        Output buffer size: 1024

Connection 2 - TCP connection from 131.108.1.27
  Client state:        CS_ACTIVE    Byte order: MSBfirst
  Unread input:        0           Unwritten output: 0
  Input buffer size:   1024        Output buffer size: 1024
```

Debugging XRemote

Although several privileged EXEC debugging commands are provided with the Cisco XRemote implementation, they are generally used for internal Cisco technical support purposes. The following is a listing of these debug commands. For each **debug** command, there is a corresponding **undebug** command to disable the reports.

debug xremote-bytestream-flow

The **debug xremote-bytestream-flow** command enables logging of XRemote bytestream state machine flow.

debug xremote-bytestream-protocol

The **debug xremote-bytestream-protocol** command enables logging of XRemote bytestream protocol packets.

debug xremote-deltas

The **debug xremote-deltas** command enables logging of XRemote delta compression information.

debug xremote-files

The **debug xremote-files** command enables logging of XRemote file access.

debug xremote-flow

The **debug xremote-flow** command enables logging of XRemote client flow.

debug xremote-replies

The **debug xremote-replies** command enables logging of XRemote replies.

debug xremote-requests

The **debug xremote-requests** command enables logging of XRemote requests.

debug xremote-packet

The **debug xremote-packet** command enables logging of XRemote packets.

debug xremote-xdm

The **debug xremote-xdm** command enables logging of XRemote XDMCP protocol and state machine events.

XRemote Global Command Summary

The following XRemote global configuration commands specify system-wide parameters for XRemote support.

[no]xremote tftp buffersize [*decimal value*]

Sets the terminal server buffer size for specific font requirements.

This **xremote** command sets the buffer size used for loading font files to a specified value (in decimal). When the X terminal requests that a font file be loaded, the terminal server must first load the font file into an internal buffer before passing it to the X terminal. The default value of 32768 is adequate for most font files, but the size can be increased as necessary for non-standard font files.

The buffer size may be set as low as 4096 bytes, and as large as the available memory on the terminal server will allow.

The command **no xremote tftp buffersize** restores the buffer size to the default of 32768 bytes.

[no] xremote tftp host [*hostname*]

Adds a specific font server as a source of fonts for the terminal (and the terminal server). Each time a new host name is entered, the list on the terminal server is updated. Font servers are queried in the order of their definition when the X terminal requests a font.

The command **no xremote tftp host** [*hostname*] removes a font server from the list maintained by the terminal server.

[no]xremote tftp retries *value*

Specifies the number of retries the font loader will attempt before declaring an error condition. The argument *value* is an integer.

The default value is 3. Under certain conditions, you may need to increase this number, particularly if the font server(s) is(are) known to be heavily loaded. Acceptable values range from 1 to 15.

The command **no xremote tftp retries** restores the retries number to the default of 3.