

**BEAM**

**Bxterm X-Terminal**

**BEAM Ltd 1995**

**Release 2.4.4**

## CONTENTS

This document is partitioned into the following sections:

<b>Section 1</b>	Introduction, outlines the contents of the BEAM Bxterm X-Terminal Software.
<b>Section 2</b>	Installation disk boot; describes how to install the Bxterm Software for disked boot X-Terminals.
<b>Section 3</b>	Installation network boot; describes how to install the Bxterm Software for network boot X-Terminals.
<b>Section 4</b>	Installation network rom boot; describes how to install the Bxterm Software for network boot X-Terminals.
<b>Section 5</b>	User Guide; describes how to use the X-Terminal and configure the optional parameters.
<b>Section 6</b>	Reference contains detailed information on the programs included.
<b>Appendix A</b>	Meaning of terms used in this manual.
<b>Appendix B</b>	Hardware compatibility.
<b>Appendix C</b>	Interactive Unix V/386 fixes.
<b>Appendix D</b>	Network Configuration with Bootp.

### Further Reference

Unix System Documentation.

X11 System Documentation

# SECTION 1

## INTRODUCTION

### Introduction

Thank you for buying one of BEAM's products. The BEAM Bxterm X-Terminal Software provides a full X-Windows, X-Terminal server with many additional features. This manual covers installation of the software for both disked and network boot Bxterm's. It also provides the information required to configure and get the most from a Bxterm.

### Information Required

During the installation of the Bxterm software a number of parameters have to be set. Recording the information before starting should save some hassle. If you do not understand any of the terms in the following table please refer to appendix A.

<b>Action</b>	<b>Information</b>	<b>Status</b>
Allocate	Host name for Bxterm	Obligatory
Allocate	Internet addresses for Bxterm	Obligatory
Record	Network Netmask	Obligatory
Record	Internet address of gateway	Optional
Record	Internet address of boot server	Obligatory network boot
Record	Internet address of login host	Optional

## SECTION 2

### INSTALLATION PROCEDURES DISKED BXTERM

#### Definition

A disked Bxterm contains a hard disk drive unit. The software is loaded onto this disk.

#### Package

The package consists of the following items:

Qty	Bxterm Host Kit component
1	Manual Box & literature
1	Bxterm software on 3.5" floppies
1	Licence Card

#### Installation Procedures

The process of installation comprises four distinct operations, adding the X-Terminal as a network node, installing the Bxterm software, setting the Licence number and configuring the required parameters.

##### Adding the Bxterm node to the network

A Host name and internet address for the X-Terminal has to be added to each host in your network that you wish to be able to communicate with the X-Terminal. The method of installation of this host name and internet address is dependent on your network configuration. The main methods used are:

##### /etc/hosts

The file **/etc/hosts** on each node has to be updated with the node name and internet address of the X-Terminal.

##### Nameserver

A selected host has a nameserver process running that provides the name to address lookup service. Two major systems are used, the Internet domain name server and the NIS name server. It is beyond the scope of this document to explain the setup of such systems. Please refer to the Unix system manuals for your system on how to add a network node.

##### Loading the Bxterm Software on a Disked Bxterm

The Bxterm software can be loaded from floppies for from a network host using NFS. If a network host is used, the Bxterm software has to be installed into a suitable directory and the host set to provide NFS read permissions on this directory for the Bxterm node.

Insert the Bxterm **Install** disk into the X-Terminal and boot the X-Terminal. If the Bxterm software has already been installed onto the hard-disk the Install system will ask if you wish to install or upgrade the software.

You should choose either the install option for a fresh install or the update option where the configuration will be retained.

The terminals configuration will now be ascertained from the the floppy disk, the Hard-disk and the Bootp network host. You will be asked if you wish to change this information. If you are using a Bootp server set the Bootp option and save the changes. The system will contact the Bootp server for this information. If you are not running a bootp server you can enter the network parameters now. Note: these values can be entered later unless you are installing the software over the network.

You will then be asked if you wish to install from floppy's or from the network. Answer **f** to this question to load from floppies or **n** to load from the network.

If you are installing the software from scratch, you will now be presented with a partition editing program to allow you to add a hard-disk partition for the software to use. Add a Bxterm partition to the hard-disk if not present. The recommend size for this partition is 100 MBytes, this leaves room for additional local clients and fonts.

If installing from floppies, insert each floppy when prompted and when complete, remove the last floppy. The system will reboot from the hard-disk.

If installing from the network, a check of the versions of software present on the network Update host will be performed. The latest release will be displayed. Answer **y** to install this release. When complete, remove the floppy. The system will reboot from the hard-disk.

### **Adding Licence Number**

On the first system boot after installation the Bxterm will present you with a dialog box asking for the serial and licence numbers. The serial and licence numbers are supplied on a licence card with the software. Use the mouse or the TAB key in conjunction with the keyboard to enter the values obtained. Once entered activate the OK button. The Xchoose program will now be displayed. If a complete Bxterm has been purchased then the serial number and licence number will have already been installed.

### **Bxterm configuration**

Before the Bxterm may access any other machine it **MUST** be given three fundamental parameters **name**, **internet address** and **network mask**.

The information can either be entered into the Bxterm system manually or can be retrieved from a **bootp** protocol server. Information on using a **bootp** server is given in a later section. Manual configuration is performed from within the xchoose program. The main menu bar of the program has a button labeled **config**. Activate the button using the mouse pointer and the left-hand mouse button. From the cascaded menu then pop-ed up select the option **Network Config**. Use the mouse or TAB key in conjunction with the Keyboard enter the required parameters.

Name	The internet host name of the Bxterm
Internet Address	The internet address of the Bxterm
Netmask	The local network mask

By default the Xserver is configured to run in VGA mode on a medium refresh rate monitor. It is recommended that you configure the Xserver for your monitor and graphics card to obtain the best performance available from the hardware. See the section on configuring the Xserver in the users guide chapter for information on how to do this.

At this stage the other parameters are optional. They may be setup if desired. Further details on configuration can be found later in this manual.

Once the Bxterm configuration has been modified press the OK Button using the mouse and the X-Terminal will self reboot to allow the changes to take effect.

## SECTION 3

### INSTALLATION PROCEDURES NETWORK BOOT

#### Definition

A network boot Bxterm performs a two stage boot procedure. An initial boot from a boot floppy, followed by a further boot across a network. Files used by the Bxterm are stored upon a Host machine designated as a boot host.

#### Package

The package consists of the following items:

Qty	Bxterm Boot Kit component
1	Manual Box & literature
2	Boot Disks
1	Bxterm software on tape or floppies
1	Licence Card

#### Installation Procedures

The process of installation comprises three distinct operations. If there are already Bxterm's installed on your network it is not necessary to perform step 1, simply configure the Bxterm, connect to the network, add the licence number, and go.

Note: It has been assumed that when adding extra terminals they are to have their boot services provided by the same host. On sites where it is wished to provide additional boot hosts the BEAM Bxterm host kit must be loaded onto each designated host. ie All three stages of installation must be performed.

##### 1. Loading the Bxterm Host Kit Software

Comprising of around 40 MBytes of information the Host kit software is loaded onto a designated host workstation. This workstation provides resources to the Bxterm both at boot and run time.

##### 2. Bxterm configuration.

The Bxterm stores network configuration information on the boot floppy disk. It is necessary to configure this information.

##### 3. Adding Licence Number

The serial number and licence number for the X-Terminal must be entered on the first boot of the X-Terminal.

#### Loading the Host Kit Software

This phase of the installation requires the presence of the System Administrator.

##### 1. Log into the boot host as the user **root**.

##### 2. Make sure there is at least 40MBytes of space on the /usr file-system. Create the bxterm directory and change to this directory.

```
mkdir /usr/bxterm
```

```
cd /usr/bxterm
```

If there is in-sufficient space on this file-system another directory can be used.

##### 3. Place tape into drive unit and load the software by the use of the following command:

```
umask 0; tar -xpf /dev/<tape>
```

(where <tape> is typically /dev/tape, /dev/rmt0 or /dev/rct0). If your tar does not support the -p option leave this out.

##### 4. The X-Terminal should now be granted NFS access to this directory. This is usually achieved by editing the /etc/exports file and perhaps running the exportfs command to update the running NFS. Normally X-Terminals are given Read/Write access. This allows users of the X-Terminals to change some parts of their configuration. In certain circumstances it is desirable to prevent users from modifying the configuration of the X-Terminals. By providing read only access the configuration of no X-terminal can be modified. ( An alternative scheme is to activate the configuration password protection facility on the Bxterm).

### Adding the Bxterm to the network

1. Add a host name and internet address for the Bxterm to your network.

Note: This procedure differs depending on the sites networking system. For example a site using no nameservers the */etc/hosts* file requires update. While on sites running NIS the NIS server requires update. It is beyond the scope of this document to explain the setup of such systems. Please refer to the Unix system manuals.

2. Set the write protect tab on one of the boot disks labeled **Boot** to allow write and insert into the Bxterm.
3. Switch on power to the Bxterm and await boot.
4. Answer the requested information in response to the prompts. The required information includes:

Prompt	Description
Node Name	Name of Bxterm on the network
Node Address	Internet address of Bxterm on the network
Netmask	Internet address mask for the network
Gateway	Internet address of Gateway machine (0.0.0.0 if none)
Server	Internet address of server machine
Server Directory	Directory on server where bxterm boot files reside

Notes: If you are unclear as to the meaning of any of the prompts please refer to appendix A.

5. The Bxterm will now boot from the server.
6. Set the write protect tab on the **Boot** diskette.

### Adding the Licence Number

On the first system boot after installation the Bxterm will present you with a dialog box asking for the serial and licence numbers. The serial number and licence number will be on a supplied licence card. Use the mouse or the TAB key in conjunction with the keyboard to enter the values obtained. Once entered activate the OK button. The Xchoose program will now be displayed. On the first boot after installation the X-Terminal will prompt for the entry of a serial and licence number. Enter the details from the information supplied on the licence card supplied with the Bxterm.

### Further configuration

The X-Terminal will present the Xchoose window. Further site specific options may now be setup as required.

By default the Xserver is configured to run in VGA mode on a medium refresh rate monitor. It is recommended that you configure the Xserver for your monitor and graphics card to obtain the best performance available from the hardware. See the section on configuring the Xserver in the users guide chapter for information on how to do this.

For further details please refer to the usage chapter.

### Further information on installation

Configuration files for the BEAM X-Terminals are stored under the directory */usr/bxterm/config*. In this directory an directory for each X-Terminal is created. The name of this directory is the internet address of the X-Terminal in hex. In each X-Terminals directory there are two configuration files. If required these can be manually modified by the system administrator. See section 6 for more information.

## SECTION 4

### INSTALLATION PROCEDURES BOOT PROM BXTERM

#### Definition

A Boot PROM Bxterm is fitted with an optional semiconductor memory PROM located on the internal Ethernet card. A Boot PROM Bxterm performs a three stage boot procedure. An initial boot from PROM followed by a further boot across a network from a Bootp server and then a further boot from a machine designated as a Bxterm boot host.

#### Package

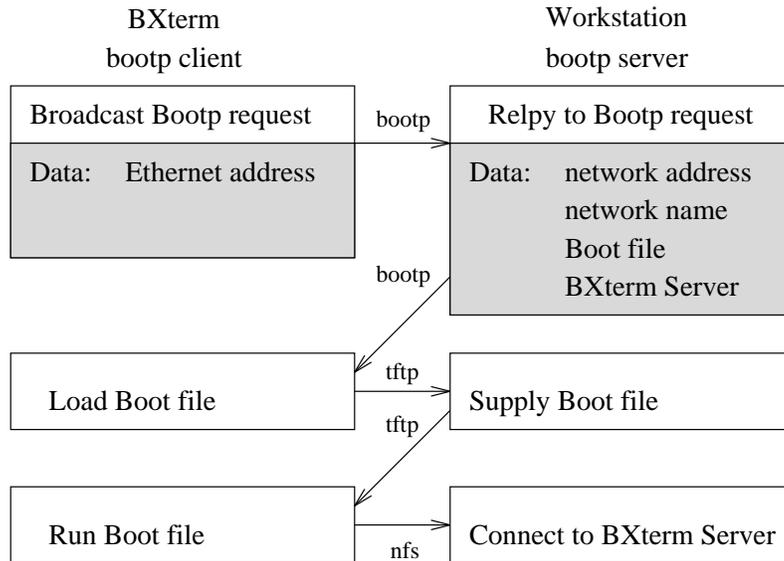
The package consists of the following items:

<b>Qty</b>	<b>Bxterm Boot Kit component</b>
1	Manual Box & literature
1	Bxterm software on tape or floppies
1	Licence Card

#### Introduction

The Boot PROM is a plug in semiconductor chip encoded with special software that allow a Bxterm to be configured without disk drives (Disk drives may optionally still be fitted). The program contained in the PROM 'talks' to a designated network host and loads all the normal Bxterm software from this host.

On power power up, a Bxterm with a Boot PROM knows nothing of its own name or network address. Its only unique identifier is its' Ethernet address built into the Ethernet card. In order to obtain more information the first thing it does is broadcast a message onto the network to say "here I am, please provide me with my identity?". This protocol is called **bootp** and is a standard Unix scheme for disk-less booting. Two machines are involved, one is called the bootp client, the other the Bootp server. The Bxterm is always a client and the server is usually a Unix Workstation. Bootp provides the Bxterm with its internet address, name and a file that is used to continue the booting process. This file is transferred to the Bxterm by another standard Unix scheme **tftp**.



## BXterm - Bootp Booting

### Installation Procedures

The process of installation comprises three distinct operations. If there are already PROM Bxterm's installed on your network it is not necessary to perform step 1, simply configure the Bxterm, connect to the network, add the licence number, and go.

Note: It has been assumed that when adding extra terminals they are to have their boot services provided by the same host. On sites where it is wished to provide additional boot hosts the BEAM Bxterm host kit must be loaded onto each designated host. ie All three stages of installation must be performed.

#### 1. Loading the Bxterm Host Kit Software

Comprising around 50 MBytes of information the Host kit software is loaded onto a designated host workstation. This workstation provides resources to the Bxterm both at boot and run time.

#### 2. Bxterm configuration

The Bxterm requires certain information during booting. This information is transferred by the **bootp** protocol. It is necessary to configure a data file on the *bootp* server before the Bxterm can boot.

#### 3. Adding Licence Number

The serial number and licence number for the X-Terminal must be entered on the first boot of the X-Terminal.

### Loading the Host Kit Software

This phase of the installation requires the presence of the System Administrator.

1. Log into the boot host as the user **root**.
2. Make sure there is at least 50MBytes of space on the /usr file-system. Create the bxterm directory and change to this directory.

```
mkdir /usr/bxterm
```

```
cd /usr/bxterm
```

If there is in-sufficient space on this file-system another directory can be used.

3. Place tape into drive unit and load the software by the use of the following command:
 

```
umask 0; tar -xpf /dev/<tape>
```

 (where <tape> is typically /dev/tape, /dev/rmt0 or /dev/rct0). If your tar does not support the -p option leave this out.
4. The X-Terminal should now be granted NFS access to this directory. This is usually achieved by editing the /etc/exports file and perhaps running the exportfs command to update the running NFS. Normally X-Terminals are given Read/Write access. This allows users of the X-Terminals to change some parts of their configuration. In certain circumstances it is desirable to prevent users from modifying the configuration of the X-Terminals. By providing read only access the configuration of the X-Terminal cannot be modified. (An alternative scheme is to activate the configuration password protection facility on the Bxterm).

### Adding the Bxterm to the network

1. Add a host name and internet address for the Bxterm to your network.

Note: This procedure differs depending on the sites networking system. For example a site using no nameservers the /etc/hosts file requires update. While on sites running NIS the NIS server requires update. It is beyond the scope of this document to explain the setup of such systems. Please refer to the Unix system manuals.

Before the PROM Bxterm can boot a networked machine must be designated a **bootp** server. It simplifies the setup process if this machine is the same machine as the Bxterm Host kit software was loaded. This host must be able to run two daemon processes:

**bootpd**            Internet Boot Protocol server  
**tftpd**            trivial file transfer program

Both of these programs are usually automatically invoked by the internet **inetd**. The configuration file for **inetd** is usually /etc/inetd.conf. Bootp and tftp are often commented out in this file, ensure that this is not the case. In the case of problems refer to the Unix system manuals.

### The /etc/bootptab file

This configuration file located on the Bootp server workstation defines the replies that the server should send when a bootp request is received.

The file comprises a number of independent entries for each client the server is expected to service. A typical entry for a Bxterm is illustrated below:

```
bxterm2:ht=1:ha=0000C0E6A6A8:\
    ip=138.44.4.102:\
    bf=/usr/bxterm/bkern_net:\
    :vm=rfc1048:\
    :T200=0x8A2C0442:
```

Note that all fields bar the first have a tag followed by '=' and a value, the ordering of the tags is unimportant, but the first item must be the name by which the Bxterm is to be known on the network. The \ indicates that the following line is a continuation of the current line.

The **bootpd** supports a large number of optional tags, those used by the Bxterm are given below:

**ha**            The **ha** tag is the Ethernet address of the Bxterm

**ht**            Hardware type, this should be set to 1 for an Ethernet network.

**ip**            The internet address of the Bxterm.

**bf**            The boot file. This tag is used to identify the file on the bootp server that will be transferred to the Bxterm during the booting sequence. The base-path, /usr/bxterm, is used to identify the location of all the files that a Bxterm uses when booted. The files need not reside on the Bootp server, another host may be used.

**vm**            The Vendor magic cookie should be set to rfc1048. This tag is only required if the Bxterm server differs from the Bootp server.

**T200**         This tag only works if the vm tag has been set to rfc1048 and is used to define the Bxterm server host. The number used is the hexadecimal encoding of the internet address of the server machine. The tag is only required if the Bxterm server differs from the Bootp server.

**Adding the Licence Number**

On the first system boot after installation the Bxterm will present you with a dialog box asking for the serial and licence numbers. The serial number and licence number will be on a supplied licence card. Use the mouse or the TAB key in conjunction with the keyboard to enter the values obtained. Once entered activate the OK button. The Xchoose program will now be displayed. On the first boot after installation the X-Terminal will prompt for the entry of a serial and licence number. Enter the details from the information supplied on the licence card supplied with the Bxterm.

**Further configuration**

The X-Terminal will present the Xchoose window. Further site specific options may now be setup as required.

By default the Xserver is configured to run in VGA mode on a medium refresh rate monitor. It is recommended that you configure the Xserver for your monitor and graphics card to obtain the best performance available from the hardware. See the section on configuring the Xserver in the users guide chapter for information on how to do this.

For further details please refer to the usage chapter.

**Further information on installation**

Configuration files for the BEAM X-Terminals are stored under the directory `/usr/bxterm/config`. In this directory an directory for each X-Terminal is created. The name of this directory is the internet address of the X-Terminal in hex. In each X-Terminals directory there are two configuration files. If required these can be manually modified by the system administrator. See section 6 for more information.

# SECTION 5

## USERS GUIDE

### Introduction

The Bxterm may be used in a number of different configurations. This section explains the various modes of operation.

### XServer Setup

By default the Xserver is configured to startup in 640x480 VGA mode with a medium refresh rate monitor. It is recommended that you configure the Xserver for your monitor and graphics card to obtain the best performance available from the hardware. To do this enter the **Xserver Config** menu entry from the **Config** item of the menu bar of the **Xchoose** application. This will present you with a form describing the current setup. Set up the Items as follows:

#### Virtual Resolution

This sets the virtual resolution of the display and hence the maximum resolution that can be displayed. By default it is set to 1152x900 which is suitable for video graphics cards with 1MByte of video memory. If however you have a video card with 2MBytes of video memory you can set this to 1280x1024.

#### Display Modes

The display modes list indicates the display resolutions that the XServer will use when the resolution hotkeys are pressed. The left most entry defines the startup resolution.

#### Monitor Type

This field defines the type of monitor connected to the X-Terminal. It is based on the performance the monitor. If the monitor is capable of running at 76Hz at 1280x1024 then the **High** monitor type should be selected. If the monitor is only capable of 60Hz at 1280x1024 or 70Hz at 1024x768 then the **Medium** monitor type should be selected. If the monitor is a lower spec than this the **Low** monitor type should be selected.

#### Mouse Type

This field sets the mouse type. By default this is set to a mouse systems mouse.

#### Font Paths

The font path list specifies the internal X-Terminal directories searched for font description files. If it is desired to have generally larger fonts on the display you can swap the order of the 75dpi and 100dpi fonts in this list.

#### Raw Config File

This entry is for special purposes only. It allows a raw edit of the XF86Config file with a text editor. It should not normally be used.

#### Reset Config File

This entry allows the configuration to be reset to factory defaults.

### Bxterm Startup

The Bxterm can be configured to startup in one of a number of modes:

**std** This is the normal default operational mode of the Bxterm. After booting the Xchoose application (Ref: Section 5). The Xchoose program allows the login to any network host and provides the ability to configure the Bxterm. This mode is typically used on sites having a number of Unix Hosts where a user often logs into

different Hosts.

- xdm** The Bxterm will connect to the default login host and present the xdm login window of this default host. The host has to have xdm running for this option to work. This mode is typically used where a user logs into the same Unix Host the Bxterm is simply an added X11 seat to the host.
- telnet** The Bxterm will connect to the default login host and present a telnet login window connected to the default host. This mode allows access to hosts that do not have xdm, typically the older generation of character based systems.
- xdmchoose**  
The Bxterm will connect to the default login host and present a remote xdm chooser window. This will only function on hosts that have the chooser version of xdm.
- local** The Bxterm will boot and run a local window manager. Remote hosts can be accessed via telnet.
- none** The Bxterm will boot as an X-Server with no local user options. A remote host can connect to the X-Server.

When the Bxterm boots if it is set to any mode other than **std** it will allow entry to the configuration system. During the booting process a prompt will be displayed. Pressing 'e' CR will boot the X-Terminal in **std** mode. The prompt has a timeout, if e is not pressed within a short time the Bxterm will boot into its configured mode.

It is also possible to change the state of the NumLock key on startup from this menu.

### Network Connection

The Bxterm can connect to the network by either Thin Ethernet (10BaseT), or AUI interface (Twisted pair 10Base2 may be purchased as an option). By default the Thin Ethernet is active. Modification of the network type is dependent on the network card installed in the system Normally the network card comes with a configuration program that allows modification of the boards parameters.

### Xauthorisation

It is possible to modify the Xauthorisation of the Bxterm. With the authorisation turned off, any host may connect to the terminal.

### Swap Size

The Bxterm has a finite amount of physical memory (Usually 8MByte). If there is a call for further memory the Bxterm can utilise the hard disk for temporary storage. This is referred to as swapping. The size of the swap area in Kilobytes is entered. A typical value for this is 8192 (8 MBytes of swap).

### Xdm - The X display Manager

Login to hosts using X-Windows is usually controlled by the X display manager program *xdm*. The *xdm* system is highly configurable and beyond the scope of this document to describe in detail for any particular host. However, its basic operation will be covered.

When, for example, a Bxterm wants to login to an xdm host, the Bxterm attempts to connect to the *xdm* process on the desired host. If a connection is accepted *xdm* will provide a Login screen asking for User name and password. Once entered, and accepted, *xdm* runs the system script file **Xsession** normally situated in the */usr/lib/X11/xdm* directory. The **Xsession** file is responsible for setting up the users environment. Normally it will run the **.xsession** or **.xinitrc** file in the users home directory. The **Xsession** file sets up a number of environment variables used by X applications.

DISPLAY	the associated display name
HOME	the home directory of the user
PATH	set to the value of DisplayManager.DISPLAY.userPath
SHELL	the users default shell
LOGNAME	the users login name
HZ	to the value DisplayManager.DISPLAY.hertz

TZ to the value DisplayManager.DISPLAY.timezone

The users `.xsession` or `.xinitrc` file typically executes a few X applications in the background and finally the Window manager in the foreground. The `xsession` login is terminated when a return from the users script file is performed. This normally occurs when the window manager is exited.

### Sun Systems

On Sun systems the `xdm` process is normally not running by default. To start up `xdm` the following command should be run

```
/usr/openwin/bin/xdm -config /usr/openwin/lib/xdm/xdm-config
```

This command is normally placed in the systems startup script.

### Bootp Protocol

The Bxterm can be configured to use a `bootp` server to acquire network configuration parameters. In order to activate this mode the `BOOTP` field in the **Network Configuration** menu of `xchoose` needs to be set to `yes`. If this parameter has been set then the Bxterm will broadcast `bootp` packets. If a `bootp` server has been configured to answer the Bxterm, it will reply with the network configuration. In order for the `bootp` protocol to provide the necessary information to the Bxterm the following information needs to be set up in the `/etc/bootptab` file on the designated `bootp` server.

Option	Required	description
ht=ether	yes	Provides information based on Ethernet address
hn	yes	Provides host name to the terminal
ip=138.44.4.101	yes	Provides IP address
ha=0000C0CE75CE	yes	Provides the Ethernet address
T200=138.44.4.1:/bxterm	optional	Provides host and directory where xvi software is installed
gw=<address>	optional	Provides the address of a gateway machine
sm=0xffff0000	optional	Network subnet mask
ns=<address>	optional	Name server address

The Ethernet hardware address can be obtained from the Bxterm either from the **Info** menu entry of `xchoose` or from the printout while the `bxterm` is broadcasting `bootp` packets.

A typical entry would be:

```
bxterm:ht=ether:hn:T200="138.44.4.1:/src3/bxterm/install":  
bxterm1:tc=bxterm:ip=138.44.4.101:ha=0000C0CE75CE:
```

### Automatic Software Update

If the `UpdateHost` field is set in the **Network Configuration** menu of `xchoose`, then the Bxterm will NFS mount the Update servers given directory add check to see if there is a later release of the software. If there is, then the user will be asked if they wish to upgrade the software. If the user answers yes then the software will be upgraded across the network.

The `UpdateHost` field should be set to the network address of the Update host followed by a ":" and then the directory in which the Bxterm release files are stored.

And example entry would be:

```
138.44.4.1:/usr/bxterm/install
```

Note that you will need the BEAM Bxterm Host Kit in order to use this option.

## **SECTION 6**

### **Introduction**

This sections describes in detail the programs supplied with the BEAM Bxterm Host Kit.

**NAME**

XF86Config - Configuration File for XFree86

**DESCRIPTION**

*XFree86* uses a configuration file called **XF86Config** for its initial setup. This configuration file is searched for in the following places:

```

/etc/XF86Config
<XRoot>/lib/X11/XF86Config.hostname
<XRoot>/lib/X11/XF86Config

```

where <XRoot> refers to the root of the X11 install tree.

This file is composed of a number of sections. Each section has the form:

```

Section "SectionName"
  SectionEntry
  ...
EndSection

```

The section names are:

- Files** (File pathnames)
- ServerFlags** (Server flags)
- Keyboard** (Keyboard configuration)
- Pointer** (Pointer configuration)
- Monitor** (Monitor description)
- Device** (Graphics device description)
- Screen** (Screen configuration)

The **Files** section is used to specify the default font path and the path to the RGB database. These paths can also be set from the command line (see *Xserver(1)*). The entries available for this section are:

**FontPath** "*path*"

sets the search path for fonts. This path is a comma separated list of directories which the X server searches for font databases. Multiple **FontPath** entries may be specified, and they will be concatenated to build up the fontpath used by the server.

X11R6 allows the X server to request fonts from a font server. A font server is specified by placing a "<trans>/<hostname>:<port\_number>" entry into the fontpath. For example, the fontpath "/usr/X11R6/lib/X11/fonts/misc,tcp/zok:7100" tells the X server to first try to locate the font in the local directory /usr/X11R6/lib/X11/fonts/misc. If that fails, then request the font from the *font server* running on machine zok listening for connections on TCP port number 7100.

**RGBPath** "*path*"

sets the path name for the RGB color database.

The **ServerFlags** section is used to specify some miscellaneous X server options. The entries available for this section are:

**NoTrapSignals**

This prevents the X server from trapping a range of unexpected fatal signals and exiting cleanly. Instead, the X server will die and drop core where the fault occurred. The default behaviour is for the X server exit cleanly, but still drop a core file. In general you never want to use this option unless you are debugging an X server problem.

**DontZap**

This disallows the use of the **Ctrl+Alt+Backspace** sequence. This sequence allows you to terminate the X server.

The **Keyboard** section is used to specify the keyboard input device, parameters and some default keyboard mapping options. The entries available for this section are:

**Protocol** "*kbd-protocol*"

*kbd-protocol* may be either **Standard** or **Xqueue**. **Xqueue** is specified when using the event queue driver

on SVR3 or SVR4.

**AutoRepeat** *delay rate*

changes the behavior of the autorepeat of the keyboard. This does not work on all platforms.

**ServerNumLock**

forces the X server to handle the numlock key internally. The X server sends a different set of keycodes for the numpad when the numlock key is active. This enables applications to make use of the numpad.

**LeftAlt** *mapping*

**RightAlt** *mapping*

**AltGr** *mapping*

**ScrollLock** *mapping*

**RightCtl** *mapping*

Allows a default mapping to be set for the above keys (note that **AltGr** is a synonym for **RightAlt**). The values that may be specified for *mapping* are:

Meta  
Compose  
ModeShift  
ModeLock  
ScrollLock  
Control

The default mapping when none of these options are specified is:

LeftAlt Meta  
RightAlt Meta  
ScrollLock Compose  
RightCtl Control

**XLeds** *led ...*

makes *led* available for clients instead of using the traditional function (Scroll Lock, Caps Lock & Num Lock). *led* is a list of numbers in the range 1 to 3.

**VTSysReq**

enables the SYSV-style VT switch sequence for non-SYSV systems which support VT switching. This sequence is Alt-SysRq followed by a function key (Fn). This prevents the X server trapping the keys used for the default VT switch sequence.

**VTInit** *"command"*

Runs *command* after the VT used by the server has been opened. The command string is passed to `"/bin/sh -c"`, and is run with the real user's id with stdin and stdout set to the VT. The purpose of this option is to allow system dependent VT initialisation commands to be run. One example is a command to disable the 2-key VT switching sequence which is the default on some systems.

The **Pointer** section is used to specify the pointer device and parameters. The entries available for this section are:

**Protocol** *"protocol-type"*

specifies the pointer device protocol type. The protocol types available are:

**BusMouse**  
**Logitech**  
**Microsoft**  
**MMSeries**  
**Mouseman**  
**MouseSystems**  
**PS/2**  
**MMHitTab**  
**Xqueue**  
**OSMouse**

One should specify **BusMouse** for the Logitech bus mouse. Also, many newer Logitech serial mice use either the **Microsoft** or **MouseMan** protocol. **Xqueue** should be specified here if it was used in the **Keyboard** section. **OSMouse** refers to the event-driver mouse interface available on SCO's SVR3. This may optionally be followed by a number specifying the number of buttons the mouse has.

**Device** *"pointer-dev"*

specifies the device the server should open for pointer input (eg, **/dev/tty00** or **/dev/mouse**). A device should not be specified when using the **Xqueue** or **OSMouse** protocols.

**BaudRate** *rate*

sets the baudrate of the serial mouse to *rate*. For mice that allow dynamic speed adjustments (like Logitech) the baudrate is changed in the mouse. Otherwise the rate is simply set on the computer's side to allow mice with non-standard rates (the standard rate is 1200).

**Emulate3Buttons**

enables the emulation of the third mouse button for mice which only have two physical buttons. The third button is emulated by pressing both buttons simultaneously.

**ChordMiddle**

handles mice which send left+right events when the middle button is used (like some Logitech Mouseman mice).

**SampleRate** *rate*

sets the number of motion/button-events the mouse sends per second. This is currently only supported for some Logitech mice.

**ClearDTR**

This option clears the DTR line on the serial port used by the mouse. This option is only valid for a mouse using the **MouseSystems** protocol. Some dual-protocol mice require DTR to be cleared to operate in MouseSystems mode. Note, in versions of XFree86 prior to 2.1, this option also cleared the RTS line. A separate **ClearRTS** option has since been added for mice which require this.

**ClearRTS**

This option clears the RTS line on the serial port used by the mouse. This option is only valid for a mouse using the **MouseSystems** protocol. Some dual-protocol mice require both DTR and RTS to be cleared to operate in MouseSystems mode. Both the **ClearDTR** and **ClearRTS** options should be used for such mice.

The **Monitor** sections are used to define the specifications of a monitor and a list of video modes suitable for use with a monitor. More than one **Monitor** section may be present in an XF86Config file. The entries available for this section are:

**Identifier** *"ID string"*

This specifies a string by which the monitor can be referred to in a later **Screen** section. Each **Monitor** section should have a unique ID string.

**VendorName** *"vendor"*

This optional entry specifies the monitor's manufacturer.

**ModelName** *"model"*

This optional entry specifies the monitor's model.

**Bandwidth** *bandwidth*

gives the bandwidth of the monitor. By default *bandwidth* is in units of MHz. It may be specified in kHz or Hz if **kHz** or **Hz** is appended to the number.

**HorizSync** *horizsync-range*

gives the range(s) of horizontal sync frequencies supported by the monitor. *horizsync-range* may be a comma separated list of either discrete values or ranges of values. A range of values is two values separated by a dash. By default the values are in units of kHz. They may be specified in MHz or Hz if **MHz** or **Hz** is added to the end of the line. The data given here is used by the X server to determine if video modes are within the specifications of the monitor. This information should be available in the monitor's handbook.

**VertRefresh** *vertrefresh-range*

gives the range(s) of vertical refresh frequencies supported by the monitor. *vertrefresh-range* may be a

comma separated list of either discrete values or ranges of values. A range of values is two values separated by a dash. By default the values are in units of Hz. They may be specified in MHz or kHz if **MHz** or **kHz** is added to the end of the line. The data given here is used by the X server to determine if video modes are within the specifications of the monitor. This information should be available in the monitor's handbook.

**Gamma** *gamma-value(s)*

This is an optional entry that can be used to specify the gamma correction for the monitor. It may be specified as either a single value or as three separate RGB values. Not all X servers are capable of using this information.

**Mode** *"name"*

indicates the start of a multi-line video mode description. The mode description is terminated with an **EndMode** line. The mode description consists of the following entries:

**DotClock** *clock*

is the dot clock rate to be used for the mode.

**HTimings** *hdisp hsyncstart hsyncend htotal*

specifies the horizontal timings for the mode.

**VTimings** *vdisp vsyncstart vsyncend vtotal*

specifies the vertical timings for the mode.

**Flags** *"flag" ...*

specifies an optional set of mode flags. **"Interlace"** indicates that the mode is interlaced. **"+HSync"** and **be used to select the polarity of the HSync signal.** **"+VSync"** and **signal.** **"CSync"** can be used to specify composite sync on hardware where this is supported.

**Modeline** *"name" mode-description*

is a single line format for specifying video modes. The *mode-description* is in four sections, the first three of which are mandatory. The first is the pixel clock. This is a single number specifying the pixel clock rate for the mode. The second section is a list of four numbers specifying the horizontal timings. These numbers are the *hdisp*, *hsyncstart*, *hsyncend*, *htotal*. The third section is a list of four numbers specifying the vertical timings. These numbers are *vdisp*, *vsyncstart*, *vsyncend*, *vtotal*. The final section is a list of flags specifying other characteristics of the mode. **Interlace** indicates that the mode is interlaced. **+HSync** and **-HSync** can be used to select the polarity of the HSync signal. **+VSync** and **-VSync** can be used to select the polarity of the VSync signal. **CSync** can be used to specify composite sync on hardware where this is supported.

The **Device** sections are used to define a graphics device (video board). More than one **Device** section may be present in an XF86Config file. The entries available for this section are:

**Identifier** *"ID string"*

This specifies a string by which the graphics device can be referred to in a later **Screen** section. Each **Device** section should have a unique ID string.

**VendorName** *"vendor"*

This optional entry specifies the graphics device's manufacturer.

**BoardName** *"model"*

This optional entry specifies the name of the graphics device.

**Chipset** *"chipset-type"*

This optional entry specifies the chipset used on the graphics board. In most cases this entry is not required because the X servers will probe the hardware to determine the chipset type.

**Ramdac** *"ramdac-type"*

This optional entry specifies the type of RAMDAC used on the graphics board. This is only used by a few of the X servers, and in most cases it is not required because the X servers will probe the hardware to determine the RAMDAC type where possible.

**DacSpeed** *speed*

This optional entry specifies the RAMDAC speed rating (which is usually printed on the RAMDAC chip). The speed is in MHz. This is only used by a few of the X servers, and only needs to be specified when the

speed rating of the RAMDAC is different from the default built in to the X server.

**Clocks** *clock ...*

specifies the dotclocks that are on your graphics board. The clocks are in MHz, and may be specified as a floating point number. The value is stored internally to the nearest kHz. The ordering of the clocks is important. It must match the order in which they are selected on the graphics board. Multiple **Clocks** lines may be specified. For boards with programmable clock chips, the **ClockChip** entry should be used instead of this. A **Clocks** entry is not mandatory for boards with non-programmable clock chips, but is highly recommended because it prevents the clock probing phase during server startup. This clock probing phase can cause problems for some monitors.

**ClockChip** "*clockchip-type*"

This optional entry is used to specify the clock chip type on graphics boards which have a programmable clock generator. Only a few X servers support programmable clock chips. For details, see the appropriate X server manual page.

**ClockProg** "*command*" [*textclock*]

This optional entry runs *command* to set the clock on the graphics board instead of using the internal code. The command string must consist of the full pathname (and no flags). When using this option, a **Clocks** entry is required to specify which clock values are to be made available to the server (up to 128 clocks may be specified). The optional *textclock* value is used to tell the server that *command* must be run to restore the textmode clock at server exit (or when VT switching). *textclock* must match one of the values in the **Clocks** entry. This parameter is required when the clock used for text mode is a programmable clock.

The command is run with the real user's id with stdin and stdout set to the graphics console device. Two arguments are passed to the command. The first is the clock frequency in MHz as a floating point number and the second is the index of the clock in the **Clocks** entry. The command should return an exit status of 0 when successful, and something in the range 1–254 otherwise.

The command is run when the initial graphics mode is set and when changing screen resolution with the hot-key sequences. If the program fails at initialisation the server exits. If it fails during a mode switch, the mode switch is aborted but the server keeps running. It is assumed that if the command fails the clock has not been changed.

**Option** "*optionstring*"

This optional entry allows the user to select certain options provided by the drivers. Multiple **Option** entries may be given. The supported values for *optionstring* are given in the appropriate X server manual pages.

**VideoRam** *mem*

This optional entry specifies the amount of videoram that is installed on the graphics board. This is measured in kBytes. In most cases this is not required because the X server probes the graphics board to determine this quantity.

**BIOSBase** *baseaddress*

This optional entry specifies the base address of the video BIOS for the VGA board. This address is normally 0xC0000, which is the default the X servers will use. Some systems, particularly those with on-board VGA hardware, have the BIOS located at an alternate address, usually 0xE0000. If your video BIOS is at an address other than 0xC0000, you must specify the base address in the XF86Config file. Note that some X servers don't access the BIOS at all, and those which do only uses the BIOS when searching for information during the hardware probe phase.

**MemBase** *baseaddress*

This optional entry specifies the memory base address of a graphics board's linear frame buffer. This entry is only used by a few X servers, and the interpretation of this base address may be different for different X servers. Refer to the appropriate X server manual page for details.

**IOBase** *baseaddress*

This optional entry specifies the IO base address. This entry is only used for a few X servers. Refer to the appropriate X server manual page for details.

**DACBase** *baseaddress*

This optional entry specifies the DAC base address. This entry is only used for a few X servers. Refer to the appropriate X server manual page for details.

**POSBASE** *baseaddress*

This optional entry specifies the POS base address. This entry is only used for a few X servers. Refer to the appropriate X server manual page for details.

**COPBASE** *baseaddress*

This optional entry specifies the coprocessor base address. This entry is only used for a few X servers. Refer to the appropriate X server manual page for details.

**Instance** *number*

This optional entry specifies the instance (which indicates of the chip is integrated on the motherboard or on an expansion card). This entry is only used for a few X servers. Refer to the appropriate X server manual page for details.

**Speedup** *"selection"*

This optional entry specifies the selection of speedups to be enabled. This entry is only used for a few X servers. Refer to the appropriate X server manual page for details.

**S3MNAdjust** *M N*

This optional entry is specific to the S3 X server. For details, refer to the *XF86\_S3(1)* manual page.

The **Screen** sections are used to specify which graphics boards and monitors will be used with a particular X server, and the configuration in which they are to be used. The entries available for this section are:

**Driver** *"driver-name"*

Each **Screen** section must begin with a **Driver** entry, and the *driver-name* given in each **Screen** section must be unique. The driver name determines which X server (or driver type within an X server when an X server supports more than one head) reads and uses a particular **Screen** section. The driver names available are:

**Accel**  
**Mono**  
**SVGA**  
**VGA2**  
**VGA16**

**Accel** is used by all the accelerated X servers (see *XF86\_Accel(1)*). **Mono** is used by the non-VGA mono drivers in the 2-bit and 4-bit X servers (see *XF86\_Mono(1)* and *XF86\_VGA16(1)*). **VGA2** and **VGA16** are used by the VGA drivers in the 2-bit and 4-bit X servers respectively. **SVGA** is used by the *XF86\_SVGA* X server.

**Device** *"device-id"*

specifies which graphics device description is to be used.

**Monitor** *"monitor-id"*

specifies which monitor description is to be used.

**ScreenNo** *scrnum*

This optional entry overrides the default screen numbering in a multi-headed configuration. The default numbering is determined by the ordering of the **Screen** sections in the *XF86Config* file. To override this, all relevant **Screen** sections must have this entry specified.

**SubSection** **"Display"**

This entry is a subsection which is used to specify some display specific parameters. This subsection is terminated by an **EndSubSection** entry. For some X servers and drivers (those requiring a list of video modes) this subsection is mandatory. For X servers which support multiple display depths, more than one **Display** subsection may be present. When multiple **Display** subsections are present, each must have a unique **Depth** entry. The entries available for the **Display** subsection are:

**Depth** *bpp*

This entry is mandatory when more than one **Display** subsection is present in a **Screen** section. When only one **Display** subsection is present, it specifies the default depth that the X server will run at. When more than one **Display** subsection is present, the depth determines which gets used by the

X server. The subsection used is the one matching the depth at which the X server is run at. Not all X servers (or drivers) support more than one depth. Permitted values for *bpp* are 8, 15, 16, 24 and 32. Not all X servers (or drivers) support all of these values. *bpp* values of 24 and 32 are treated equivalently by those X servers which support them.

**Weight** *RGB*

This optional entry specifies the relative RGB weighting to be used for an X server running at 16bpp. This may also be specified from the command line (see *XFree86(1)*). Values supported by most 16bpp X servers are **555** and **565**. For further details, refer to the appropriate X server manual page.

**Virtual** *xdim ydim*

This optional entry specifies the virtual screen resolution to be used. *xdim* must be a multiple of either 8 or 16 for most colour X servers, and a multiple of 32 for the monochrome X server. The given value will be rounded down if this is not the case. For most X servers, video modes which are too large for the specified virtual size will be rejected. If this entry is not present, the virtual screen resolution will be set to accommodate all the valid video modes given in the **Modes** entry. Some X servers do not support this entry. Refer to the appropriate X server manual pages for details.

**ViewPort** *x0 y0*

This optional entry sets the upper left corner of the initial display. This is only relevant when the virtual screen resolution is different from the resolution of the initial video mode. If this entry is not given, then the initial display will be centered in the virtual display area.

**Modes** "*modename*" ...

This entry is mandatory for most X servers, and it specifies the list of video modes to use. The video mode names must correspond to those specified in the appropriate **Monitor** section. Most X servers will delete modes from this list which don't satisfy various requirements. The first valid mode in this list will be the default display mode for startup. The list of valid modes is converted internally into a circular list. It is possible to switch to the next mode with **Ctrl+Alt+Keypad-Plus** and to the previous mode with **Ctrl+Alt+Keypad-Minus**.

**Visual** "*visual-name*"

This optional entry sets the default root visual type. This may also be specified from the command line (see *Xserver(1)*). The visual types available for 8bpp X servers are (default is **PseudoColor**):

**StaticGray**  
**GrayScale**  
**StaticColor**  
**PseudoColor**  
**TrueColor**  
**DirectColor**

The visual type available for the 16bpp and 32bpp X servers is **TrueColor**.

The visual type available for the 1bpp X server is **StaticGray**.

The visual types available for the 4bpp X server are (default is **PseudoColor**):

**StaticGray**  
**GrayScale**  
**StaticColor**  
**PseudoColor**

**Option** "*optionstring*"

This optional entry allows the user to select certain options provided by the drivers. Multiple **Option** entries may be given. The supported values for *optionstring* are given in the appropriate X server manual pages.

**Black** *red green blue*

This optional entry allows the "black" colour to be specified. This is only supported with the VGA2 driver in the *XF86\_Mono* server (for details see *XF86\_Mono(1)*).

**White** *red green blue*

This optional entry allows the "white" colour to be specified. This is only supported with the VGA2

driver in the XF86\_Mono server (for details see *XF86\_Mono(1)*).

For an example of an XF86Config file, see the file installed as <XRoot>/lib/X11/XF86Config.eg.

## FILES

/etc/XF86Config  
<XRoot>/lib/X11/XF86Config.*hostname*  
<XRoot>/lib/X11/XF86Config

Note: <XRoot> refers to the root of the X11 install tree.

## SEE ALSO

X(1), Xserver(1), XFree86(1), XF86\_SVGA(1), XF86\_VGA16(1), XF86\_Mono(1), XF86\_S3(1), XF86\_8514(1), XF86\_Mach8(1), XF86\_Mach32(1), XF86\_P9000(1), XF86\_AGX(1), XF86\_W32(1).

## AUTHORS

Refer to the *XFree86(1)* manual page.

**NAME**

XFree86 - X11R6 for UNIX on x86 platforms

**DESCRIPTION**

XFree86 is a collection of X servers for UNIX-like OSs on Intel x86 platforms. This work is derived from X386 1.2 which was contributed to X11R5 by Snitily Graphics Consulting Service.

**CONFIGURATIONS**

XFree86 operates under the following operating systems:

- SVR3.2: SCO 3.2.2, 3.2.4
- SVR4.0: ESIX, Microport, Dell, UHC, Consensys, MST, ISC, AT&T, NCR
- SVR4.2: Consensys, Univel (UnixWare)
- Solaris (x86) 2.1, 2.4
- FreeBSD 1.1.5, NetBSD 1.0 (i386 port only)
- BSD/386 version 1.1
- Mach (from CMU)
- Linux
- Amoeba version 5.1
- Minix-386vm version 1.6.25.1

**NETWORK CONNECTIONS**

XFree86 supports connections made using the following reliable byte-streams:

*Local*

XFree86 supports local connections via Streams pipe via various mechanisms, using the following paths (*n* represents the display number):

- /dev/X/server.*n* (SVR3 and SVR4)
- /dev/X/Nserver.*n* (SVR4)
- /dev/X*n*S and /dev/X*n*R (SCO SVR3)

On SVR4.0.4, if the *Advanced Compatibility Package* is installed, and in SVR4.2, XFree86 supports local connections from clients for SCO XSight/ODT, and (with modifications to the binary) clients for ISC SVR3.

*Unix Domain*

XFree86 uses */tmp/.X11-unix/Xn* as the filename for the socket, where *n* is the display number.

*TCPIP*

XFree86 listens on port *htons(6000+n)*, where *n* is the display number.

*Amoeba RPC*

This is the default communication medium used under native Amoeba. Note that under Amoeba, the server should be started with a "*hostname:displaynumber*" argument.

**ENVIRONMENT VARIABLES**

For operating systems that support local connections other than Unix Domain sockets (SVR3 and SVR4), there is a compiled-in list specifying the order in which local connections should be attempted. This list can be overridden by the *XLOCAL* environment variable described below. If the display name indicates a best-choice connection should be made (e.g. *:0.0*), each connection mechanism is tried until a connection succeeds or no more mechanisms are available. Note: for these OSs, the Unix Domain socket connection is treated differently from the other local connection types. To use it the connection must be made to *unix:0.0*.

The *XLOCAL* environment variable should contain a list of one more more of the following:

- NAMED
- PTS
- SCO
- ISC

which represent SVR4 Named Streams pipe, Old-style USL Streams pipe, SCO XSight Streams pipe, and ISC Streams pipe, respectively. You can select a single mechanism (e.g. *XLOCAL=NAMED*), or an ordered list (e.g. *XLOCAL="NAMED:PTS:SCO"*). This variable overrides the compiled-in defaults. For SVR4 it is recommended that *NAMED* be the first preference connection. The default setting is *PTS:NAMED:ISC:SCO*.

To globally override the compiled-in defaults, you should define (and export if using *sh* or *ksh*) *XLOCAL* globally. If you use *startx/xinit*, the definition should be at the top of your *.xinitrc* file. If you use *xdm*, the definitions should be early on in the *<XRoot>/lib/X11/xdm/Xsession* script.

## OPTIONS

In addition to the normal server options described in the *Xserver(1)* manual page, *XFree86* accepts the following command line switches:

**vtXX** *XX* specifies the Virtual Terminal device number which *XFree86* will use. Without this option, *XFree86* will pick the first available Virtual Terminal that it can locate. This option applies only to SVR3, SVR4, Linux, and BSD OSs with the 'syscons' or 'pcvt' driver.

**-probeonly**

Causes the server to exit after the device probing stage. The XF86Config file is still used when this option is given, so information that can be auto-detected should be commented out.

**-quiet** Suppress most informational messages at startup.

**-bpp n** Set number of bits per pixel. The default is 8. Legal values are 8, 15, 16, 24, 32. Not all servers support all values.

**-weight nnn**

Set RGB weighting at 16 bpp. The default is 565. This applies only to those servers which support 16 bpp.

**-gamma value**

Set the gamma correction. *value* must be between 0.1 and 10. The default is 1.0 This value is applied equally to the R, G and B values. Not all servers support this.

**-rgamma value**

Set the red gamma correction. *value* must be between 0.1 and 10. The default is 1.0 Not all servers support this.

**-ggamma value**

Set the green gamma correction. *value* must be between 0.1 and 10. The default is 1.0 Not all servers support this.

**-bgamma value**

Set the blue gamma correction. *value* must be between 0.1 and 10. The default is 1.0 Not all servers support this.

**-showconfi g**

Print out a list of screen drivers configured in the server.

**-verbose**

Maximise information printed at startup (this is the default).

**-xf86confi g file**

Read the server configuration from *file*. This option is only available when the server is run as root (i.e. with real-uid 0).

**-keeppty**

Prevent the server from detaching its initial controlling terminal. This option is only useful when debugging the server.

## KEYBOARD

Multiple key presses recognized directly by *XFree86* are:

**Ctrl+Alt+Backspace**

Immediately kills the server -- no questions asked. (Can be disabled by specifying "DontZap" in the **ServerFlags** section of the XF86Config file.)

**Ctrl+Alt+Keypad-Plus**

Change video mode to next one specified in the configuration file, (increasing video resolution order).

**Ctrl+Alt+Keypad-Minus**

Change video mode to previous one specified in the configuration file, (decreasing video resolution order).

**Ctrl+Alt+F1...F12**

For BSD systems using the syscons driver and Linux, these keystroke combinations are used to switch to Virtual Console 1 through 12.

## SETUP

*XFree86* uses a configuration file called **XF86Config** for its initial setup. Refer to the *XF86Config(4/5)* manual page for more information.

## FILES

<XRoot>/bin/XF86_SVGA	The color SVGA X server
<XRoot>/bin/XF86_Mono	The monochrome X server for VGA and other mono cards
<XRoot>/bin/XF86_S3	The accelerated S3 X server
<XRoot>/bin/XF86_Mach8	The accelerated Mach8 X server
<XRoot>/bin/XF86_Mach32	The accelerated Mach32 X server
<XRoot>/bin/XF86_P9000	The accelerated P9000 X server
<XRoot>/bin/XF86_AGX	The accelerated AGX X server
<XRoot>/bin/XF86_W32	The accelerated ET4000/W32 X server
<XRoot>/bin/XF86_8514	The accelerated 8514/A X server
/etc/XF86Config	Server configuration file
<XRoot>/lib/X11/XF86Config. <i>hostname</i>	Server configuration file
<XRoot>/lib/X11/XF86Config	Server configuration file
<XRoot>/bin/*	Client binaries
<XRoot>/include/*	Header files
<XRoot>/lib/*	Libraries
<XRoot>/lib/X11/fonts/*	Fonts
<XRoot>/lib/X11/rgb.txt	Color names to RGB mapping
<XRoot>/lib/X11/XErrorDB	Client error message database
<XRoot>/lib/X11/app-defaults/*	Client resource specifications
<XRoot>/man/man?/*	Manual pages
/etc/Xn.hosts	Initial access control list for display <i>n</i>

Note: <XRoot> refers to the root of the X11 install tree.

## SEE ALSO

X(1), Xserver(1), xdm(1), xinit(1), Xconfig(4/5), XF86\_SVGA(1), XF86\_VGA16(1), XF86\_Mono(1), XF86\_Accel(1)

## AUTHORS

For X11R5, *XF86 1.2* was provided by:

Thomas Roell, [roell@informatik.tu-muenchen.de](mailto:roell@informatik.tu-muenchen.de)  
TU-Muenchen: Server and SVR4 stuff

Mark W. Snitily, [mark@sgcs.com](mailto:mark@sgcs.com)  
SGCS: SVR3 support, X Consortium Sponsor

... and many more people out there on the net who helped with ideas and bug-fixes.

XFree86 was integrated into X11R6 by the following team:

Stuart Anderson [Stuart.Anderson@ColumbiaSC.NCR.COM](mailto:Stuart.Anderson@ColumbiaSC.NCR.COM)  
Doug Anson [danson@lgc.com](mailto:danson@lgc.com)  
Gertjan Akkerman [akkerman@dutiba.twi.tudelft.nl](mailto:akkerman@dutiba.twi.tudelft.nl)  
Mike Bernson [mike@mbsun.mlb.org](mailto:mike@mbsun.mlb.org)  
Robin Cutshaw [robin@XFree86.org](mailto:robin@XFree86.org)

David Dawes     *dawes@XFree86.org*  
 Marc Evans     *marc@XFree86.org*  
 Pascal Haible   *haible@izfm.uni-stuttgart.de*  
 Matthieu Herrb   *Matthieu.Herrb@laas.fr*  
 Dirk Hohndel    *hohndel@XFree86.org*  
 David Holland    *davidh@use.com*  
 Alan Hourihane   *alanh@fairlite.demon.co.uk*  
 Jeffrey Hsu     *hsu@soda.berkeley.edu*  
 Glenn Lai       *glenn@cs.utexas.edu*  
 Ted Lemon       *mellon@ncd.com*  
 Rich Murphey    *rich@XFree86.org*  
 Hans Nasten     *nasten@everyware.se*  
 Mark Snitily     *mark@sgcs.com*  
 Randy Terbush    *randyt@cse.unl.edu*  
 Jon Tombs       *tombs@XFree86.org*  
 Kees Verstoep    *versto@cs.vu.nl*  
 Paul Vixie       *paul@vix.com*  
 Mark Weaver     *Mark\_Weaver@brown.edu*  
 David Wexelblat *dwex@XFree86.org*  
 Philip Wheatley *Philip.Wheatley@ColumbiaSC.NCR.COM*  
 Thomas Wolfram   *wolf@prz.tu-berlin.de*  
 Orest Zborowski *orestz@eskimo.com*

The *XFree86* enhancement package was provided by:

David Dawes,     *dawes@XFree86.org*  
                   Administration of FTP repository and mailing lists. Source tree management and integration, accelerated server integration, fixing, and coding.

Glenn Lai,       *glenn@cs.utexas.edu*  
                   The SpeedUp code for ET4000 based SVGA cards, and ET4000/W32 accelerated server.

Jim Tsillas,     *jtsilla@ccs.neu.edu*  
                   Many server speedups from the fXF86 series of enhancements.

David Wexelblat, *dwex@XFree86.org*  
                   Release coordination, integration of the fXF86 code into the default server, many driver fixes, and driver documentation, assembly of the VGA card/monitor database, development of the generic video mode listing. Accelerated server integration, fixing, and coding.

Dirk Hohndel,    *hohndel@XFree86.org*  
                   Linux shared libraries and release coordination. Accelerated server integration and fixing. Generic administrivia and documentation.

Amancio Hasty Jr., *hasty@netcom.com*  
                   Porting to **386BSD** version 0.1 and XS3 development.

Rich Murphey,    *rich@XFree86.org*  
                   Ported to **386BSD** version 0.1 based on the original port by Pace Willison. Support for **386BSD**, **FreeBSD**, and **NetBSD**.

Robert Baron,    *Robert.Baron@ernst.mach.cs.cmu.edu*  
                   Ported to **Mach**.

Orest Zborowski, *orestz@eskimo.com*  
                   Ported to **Linux**.

Doug Anson,     *danson@lgc.com*  
                   Ported to **Solaris x86**.

David Holland,    *davidh@use.com*  
                   Ported to **Solaris x86**.

David McCullough, *davidm@stallion.oz.au*  
                   Ported to **SCO SVR3**.

Kees Verstoep, *versto@cs.vu.nl*  
Ported to **Amoeba** based on Leendert van Doorn's original Amoeba port of X11R5.

Marc Evans, *Marc@XFree86.org*  
Ported to **OSF/1**.

Philip Homburg, *philip@cs.vu.nl*  
Ported to **Minix-386vm**.

Jon Tombs, *tombs@XFree86.org*  
S3 server and accelerated server coordination.

Harald Koenig, *koenig@tat.physik.uni-tuebingen.de*  
S3 server development.

Bernhard Bender, *br@elsa.mhs.compuserve.com*  
S3 server development.

Kevin Martin, *martin@cs.unc.edu*  
Overall work on the base accelerated servers (ATI and 8514/A).

Rik Faith, *faith@cs.unc.edu*  
Overall work on the base accelerated servers (ATI and 8514/A).

Tiago Gons, *tiago@comosjn.hobby.nl*  
Mach8 and 8514/A server development

Hans Nasten, *nasten@everyware.se*  
Mach8, 8514/A, and S3 server development and BSD/386 support

Mike Bernson, *mike@mbsun.mlb.org*  
Mach32 server development.

Mark Weaver, *Mark\_Weaver@brown.edu*  
Mach32 server development.

Craig Groeschel, *craig@metrolink.com*  
Mach32 server development.

Henry Worth, *Henry.Worth@amail.amdahl.com*  
AGX server.

Erik Nygren, *nygren@mit.edu*  
P9000 server.

Harry Langenbacher *harry@brain.jpl.nasa.gov*  
P9000 server.

Chris Mason, *mason@mail.csh.rit.edu*  
P9000 server.

Simon Cooper, *scooper@vizlab.rutgers.edu*  
Cirrus accelerated code (based on work by Bill Reynolds).

Harm Hanemaayer, *hhanemaa@cs.ruu.nl*  
Cirrus accelerated code.

Mike Tierney, *floyd@eng.umd.edu*  
WD accelerated code.

Bill Conn, *conn@bnr.ca*  
WD accelerated code.

Paolo Severini, *lendl@dist.dist.unige.it*  
AL2101 SVGA driver

Manfred Brands *mb@oceanics.nl*  
Cirrus 6420 SVGA driver

Frank Dikker *dikker@cs.utwente.nl*  
MX SVGA driver

Craig Struble [cstruble@acm.vt.edu](mailto:cstruble@acm.vt.edu)  
Video7 SVGA driver

Gertjan Akkerman, [akkerman@dutiba.twi.tudelft.nl](mailto:akkerman@dutiba.twi.tudelft.nl)  
16 colour VGA server, and XF86Config parser.

Davor Matic, [dmatic@Athena.MIT.EDU](mailto:dmatic@Athena.MIT.EDU)  
Hercules driver.

Pascal Haible, [haible@izfm.uni-stuttgart.de](mailto:haible@izfm.uni-stuttgart.de)  
Banked monochrome VGA support, Hercules support, and mono frame buffer support for dumb monochrome devices

... and many more people out there on the net who helped with beta-testing this enhancement.

*XFree86* source is available from the FTP server <ftp.XFree86.org>, among others. Send email to [XFree86@XFree86.org](mailto:XFree86@XFree86.org) for details.

**NAME**

XF86\_SVGA - non-accelerated SVGA X Window System servers for UNIX on x86 platforms

**SYNOPSIS**

**XF86\_SVGA** [:displaynumber] [ option ] ...

**DESCRIPTION**

*XF86\_SVGA* is an 8-bit PseudoColor, 16-bit TrueColor and 24-bit TrueColor server for Super-VGA cards. It is derived from the *X386* server provided with X11R5. Note: 16-bit and 24-bit TrueColor is currently only supported for some Cirrus chips.

**CONFIGURATIONS**

The *XF86\_SVGA* server supports the following popular SuperVGA chipsets in 256 color mode. Virtual resolutions up to (approximately) 1152x900 are supported, using (up to) 1Mb of display memory. The Western Digital WD90C33 and some of the Cirrus chipsets support up to 2Mb of display memory and virtual resolutions of 1280x1024 and higher. Some of the Cirrus chipsets also support 16bpp and 32bpp (truecolor) modes on certain configurations. Generic VGA cards are also supported at 8bpp 320x200 only.

## ATI:

18800, 18800-1, 28800-2, 28800-4, 28800-5, 28800-6, 68800-3, 68800-6, 68800AX, 68800LX, 88800

## Tseng:

ET3000, ET4000, ET4000/W32

## Western Digital:

PVGA1, WD90C00, WD90C10, WD90C11, WD90C30, WD90C31, WD90C33

## Genoa:

GVGA

## Trident:

TVGA8800CS, TVGA8900B, TVGA8900C, TVGA8900CL, TVGA9000

## NCR:

77C22, 77C22E

## Cirrus Logic:

CLGD5420, CLGD5422, CLGD5424, CLGD5426, CLGD5428, CLGD5429, CLGD5430, CLGD5434, CLGD6205, CLGD6215, CLGD6225, CLGD6235, CL6420

## Compaq:

AVGA

## Oak:

OTI067, OTI077

## Advance Logic:

AL2101

## MX:

MX68000, MX68010

## Video7:

HT216-32

Accelerated support is included for most of the Cirrus chipsets, and for the Western Digital WD90C31 and WD90C33 chipsets. Accelerated support for the ET4000/W32 is implemented in a separate server (see *XF86\_W32(1)*). Users of boards based on ATI's Mach8 and Mach32 chipsets should refer to the *XF86\_Mach8(1)* and *XF86\_Mach32(1)* manual pages, respectively.

**OPTIONS**

In addition to the normal server options described in the *Xserver(1)* manual page, *XF86\_SVGA* accepts some more command line switches, as described in the *XFree86(1)* manpage.

## SETUP

*XFree86* uses a configuration file called **XF86Config** for its initial setup. See the *XF86Config(4/5)* manpage for general details. Here only the *XF86\_SVGA* specific parts are explained.

This server requires a **Screen** section in the *XF86Config* file with the **Driver** entry set to **svga**.

Entries for the **Device** section in the *XF86Config* file include:

**chipset** "*name*"

specifies a chipset so the correct driver can be used. Possible chipsets are:

ATI:

vgawonder

Tseng:

et3000, et4000, et4000w32, et4000w32i, et4000w32p

Western Digital:

pvga1, wd90c00, wd90c10, wd90c30, wd90c31, wd90c33

Genoa:

gvga

Trident:

tvga8800cs, tvga8900b, tvga8900c, tvga8900cl, tvga9000

NCR:

ncr77c22, ncr77c22e

Cirrus Logic:

clgd5420, clgd5422, clgd5424, clgd5426, clgd5428, clgd5429, clgd5430, clgd5434, clgd6205, clgd6215, clgd6225, clgd6235, cl6420

Compaq:

cpq\_avga

OAK:

oti067, oti077

Advance Logic:

al2101

MX:

mx

Video7:

video7

Generic:

generic

**Option** "*optionstring*"

allows the user to select certain options provided by the drivers. Currently the following strings are recognized:

**legend** - for Sigma Legend ET4000-based boards. This option enables a special clock-selection algorithm used on Legend boards, and **MUST** be specified for these boards to function correctly.

**swap\_hibit** - for Western Digital/PVGA1 chipsets. Some Western Digital based boards require the high-order clock-select lead to be inverted. It is not possible for the server to determine this information at run-time. If the 9th clock in the list of clocks detected by the server is less than 30Mhz, this option likely needs to be set.

**hibit\_low**, **hibit\_high** - for Tseng ET4000 chipsets. With some ET4000 cards, the server has difficulty getting the state of the high-order clocks select bit right when started from a high-resolution text mode. These options allow the correct initial state of that bit to be specified. To find out what the correct initial state is, start the server from an 80x25 text mode. This option is only needed if the clocks reported by the

server when started from a high-resolution text mode differ from those reported when it is started from an 80x25 text mode.

**8clocks** - for the PVGA1 chipset the default is 4 clocks. Some cards with this chipset may support 8 clocks. Specifying this option will allow the driver to detect and use the extra clocks.

**16clocks** - for Trident TVGA8900B and 8900C chipsets. Some newer boards using 8900B and 8900C chipsets actually support 16 clocks rather than the standard 8 clocks. Such boards will have a "TCK9002" or "TCK9004" chip on them. Specifying this option will allow the driver to detect and use the extra 8 clocks.

**probe\_clocks** - for Cirrus chipsets. The Cirrus driver has a fixed set of clocks that are normally used. Specifying this option will force the driver to probe for clocks instead of reporting the built-in defaults. This option is for debugging purposes only.

**power\_saver** - This option enables the server to use the power saving feature of "green" monitors instead of blanking when the screen saver is activated. This option is experimental.

**noaccel** - for Cirrus and WD chipsets. This option disables the accelerated features for the clgd5426, clgd5428, wd90c31 and wd90c33 chipsets.

**fi fo\_conservative** - for Cirrus chipsets. This option sets the CRT FIFO threshold to a conservative value for dot clocks above 65MHz. This reduces performance, but may help in eliminating problems with "streaks" on the screen during BitBLT operations

**fi fo\_aggressive** - for Cirrus chipsets. This option sets the CRT FIFO threshold to an aggressive value for dot clocks above 65MHz. This may increase performance.

**slow\_dram** - for Cirrus chipsets. This option sets the DRAM timings for slow DRAM chips.

**fast\_dram** - for ET4000 and Cirrus chipsets. This option sets the DRAM timings for fast DRAM chips.

**no\_2mb\_banksel** - for Cirrus chipsets. This option is required for Cirrus cards with 2MB of videoram which is in the form of 512kx8 DRAMs (4 chips) rather than 256kx4 DRAMs (16 chips).

**no\_bitblt** - for Cirrus chipsets. This option disables use of hardware BitBLT.

**linear** - Attempt a linear mapping of the framebuffer into high memory. Currently only supported for some Cirrus configurations.

**med\_dram, favour\_bitblt, sw\_cursor, clgd6225\_lcd, mmio** - more Cirrus-specific options. Refer to /usr/X11R6/lib/X11/doc/README.cirrus for a detailed description of Cirrus options.

**speedup** "*selection*"

sets the selection of SpeedUps to use. The optional selection string can take the following values:

none  
all

If the selection string is omitted, or if the **speedup** option is omitted, the selection defaults to "all". Some of the SpeedUps can only be used with the ET4000, WD90C31 and WD90C33 chipsets and others requires a virtual resolution with a xdim of 1024. SpeedUps that won't work with a given configuration are automatically disabled.

**nospeedup**

disables the SpeedUp code. This is equivalent to **speedup "none"**.

Note that *XFree86* has some internal capabilities to determine what hardware it is running on. Thus normally the keywords *chipset*, *clocks*, and *videoram* don't have to be specified. But there may be occasions when this autodetection mechanism fails, (for example, too high of load on the machine when you start the server). For cases like this,

one should first run *XF86\_SVGA* on an unloaded machine, look at the results of the autodetection (that are printed out during server startup) and then explicitly specify these parameters in the configuration file. **It is recommended that all parameters, especially Clock values, be specified in the XF86Config file.**

## FILES

<XRoot>/bin/XF86_SVGA	The SVGA color X server
/etc/XF86Config	Server configuration file
<XRoot>/lib/X11/XF86Config	Server configuration file
<XRoot>/lib/X11/doc/README.at	Extra documentation for the ATI vga wonder driver
<XRoot>/lib/X11/doc/README.cirrus	Extra documentation for the Cirrus driver
<XRoot>/lib/X11/doc/README.trident	Extra documentation for the Trident driver
<XRoot>/lib/X11/doc/README.tseng	Extra documentation for the ET4000 and ET3000 drivers
<XRoot>/lib/X11/doc/README.Video7	Extra documentation for the Video7 driver
<XRoot>/lib/X11/doc/README.WstDig	Extra documentation for the WD/PVGA driver

Note: <XRoot> refers to the root of the X11 install tree.

## SEE ALSO

X(1), Xserver(1), XFree86(1), XF86Config(4/5), xdm(1), xinit(1)

## BUGS

There are no known bugs at this time, although we welcome reports emailed to the address listed below.

## CONTACT INFO

*XFree86* source is available from the FTP servers *ftp.XFree86.org*. Send email to *XFree86@XFree86.org* for details.

## AUTHORS

Refer to the *XFree86(1)* manual page.

## NAME

xchoose – BEAM Bxterm host chooser and configuration application

## SYNOPSIS

**xchoose** [ options ]

## DESCRIPTION

*Xchoose* is an X-Windows/Motif application that provides remote login and X-Terminal configuration facilities for the Bxterm. As supplied Xchoose is the initial interface displayed to the user when the Bxterm is booted.

The *xchoose* application consists of a menu bar at the top with a list of hosts below it. Operations are invoked by selecting an item from the top menu bar. Selection is performed with the mouse and left hand mouse button. The main menu bar provides the following operations :-

<b>Login</b>	Perform a login attempt on the host selected in the host list.
<b>Manual Login</b>	Perform a login to a user specified host using either xdm or Telnet protocols. When this operation is selected a dialog box will be presented requesting further information, the host name or internet address and the mode of operation xdm or telnet. It is only possible to use xdm if the specified host is configured to provide xdm logins. It is possible to use the name of the host only if the Bxterm has been setup with the internet address of a configured nameserver host.
<b>Local</b>	Startup a local X window session. This starts up a local window manager and allows both local and remote X-Clients to be run. Remote clients can be run using either the telnet or rexec systems.
<b>Config</b>	Configure the Bxterm.
<b>Help</b>	Online help system.

## Bxterm Configuration

Configuration of the Bxterm is performed by commands accessed via the **config** main menu button. The configuration options include:

### Add Host, Delete Host

The *xchoose* application maintains a list of hosts that a user may quickly log into. The list of hosts configured appear below the main menu bar. To add a host to this list some information is required. The name of the host, its internet address and the method of login. The name and address are self explanatory, the method of login may be set to one of two options. **xdm** or **telnet** Select **xdm** for hosts that support and are configured to run the xdm (X display manager) protocol. An **xdm** login typically presents the user with a login prompt on the X-terminal and once logged in a defined X environment. A **telnet** login by contrast can be used to all hosts that support the telnet protocol. The Bxterm invokes a basic unmanaged test terminal window for the login. Once logged into the remote system the user may run a window manager or X application in the normal manner.

**Startup Config** This option will present a dialog for editing the Bxterm's startup parameters.

**Network Config** This option will present a dialog for editing the Bxterm's network parameters.

**Xserver Config** This option will present a dialog for editing the Bxterm's Xserver parameters.

**Printer Config** This option will present a dialog for editing the Bxterm's printer parameters.

**Load Fonts** This option will present a dialog for adding fonts to the Bxterm's internal list.

**Set Boot** Only applicable to Bxterm's that have MSDOS installed. This option defines the default booting mode of the Terminal. **Bxterm only** allows the system to boot only as an X-Terminal.

**MSDOS & Bxterm** the primary boot is by default into MSDOS. **Bxterm & MSDOS** the primary boot in this case is as a Terminal.

However if during the boot sequence the user types ? early in the boot process the system to be booted can be manually selected.

**Reboot** Performs a reboot of the Bxterm.

**Set Password** Access to the configuration options can be password protected. Select this option to setup a password. To clear a password use enter password followed by set password where the password entered is an empty string.

**Enter Password** This option must be selected prior to any configuration if a password has been set.

### Startup Configuration

This dialog provides the ability to edit the Bxterm's startup parameters.

#### Mode

This parameter sets up the boot mode of the Bxterm. It can be set to one of the following options:

- "std" Standard boot. Starts up with the xchoose application allowing the user to choose a login host.
- "xdm" Boot to give Xdm login prompt to specified default host. This requires the configuration of Xdm on the specified host.
- "telnet" Boot to give telnet login to the login host.
- "xchoose" Boot to Xdm remote chooser window. Require setup of Xdm on a network host.
- "local" Boot as a stand alone Host. A local window manager is invoked. Login to a remote host must be performed via telnet from within the pop-up menus of the window manager.
- "none" Boot to raw X-server. Only the X-server is invoked. Remote hosts can access the display in the normal manner ( -display <Bxterm name:0 or internet address:0>).

#### Window Manager

This parameter defines the Window Manager used in local mode. It can be either **olvwm**, **twm** or **mwm**.

#### X Authorisation

This option turns the X-Authorisation system on/off

**Swap Size** This option, only available to disked Bxterm's, offers the ability to set up a local swap space on the local hard disk. An integer value should be entered which is the size in KBytes of the swap space. This is only needed in specialised applications that make large usage of the XServer's memory resources. A suitable size would be 4096 KBytes.

**NumLock** This parameter changes the state of the NumLock key on startup.

### Network Configuration

This dialog provides the ability to edit the Bxterm's network parameters.

**Bootp** Configures the Bxterm to use the **bootp** protocol to acquire the network configuration.

**Name** Host name for the Bxterm.

#### Internet Address

Enter the internet address of this Bxterm. The address should be entered in normal dot notation. For example :-  
137.44.4.78

145.65.126.145

### Network Mask

Enter the network mask in internet address format. The network class A,B or C defines these types of network address.

Class	Netmask	Network Type
A	255.0.0.0	1 Byte Network, 3 Bytes Host Id
B	255.255.0.0	2 Byte Network, 2 Bytes Host Id
C	255.255.255.255.0	3 Byte Network, 1 Byte Host Id

### Gateway

Enter the internet address of a host on the local network providing gateway service to other networks. This file is optional. If you do not have a gateway do not enter a value.

### Nameserver

Enter the internet address of the a host on your network that provides DNS name service ( DNS - Domain name server).

**Host** This is the internet address or host name (if the nameserver parameter has been set up) of the host to use for login with **x**dm or **telnet** boot modes.

### Update Host

This is the internet address or host name (if the nameserver parameter has been set up) of the host to use for performing the automatic update of Bxterm software with the directory on the host where the software is installed. An example entry would be:

138.44.4.1:/usr/bxterm/install

Note that you will need the BEAM Bxterm Host Kit in order to use this option.

## XServer Configuration

This dialog allows you to modify the XServer's parameters. For more information see the manual on XF86Config.

### Virtual resolution

This parameter sets up the size of the Virtual display that the XServer uses. It is normally set to the highest resolution that your graphics board will support. This is "1152 900" on 1Mbyte graphics boards and "1280 1024" on 2MByte graphics boards.

### Display Modes

This list of parameters define the list of available screen display modes selectable with the resolution set hot key sequence. The left most resolution is the default power up resolution.

### Monitor Type

This defines the type of monitor connected to the system. Valid monitor types are:

**Low** This option defines a low scan rate monitor. Monitors that are only capable of 60Hz Vertical refresh rate at 1024x768 should use this mode.

**Medium** This option defines a medium scan rate monitor. Monitors that are capable of 70Hz Vertical refresh rate at 1024x768 and/or 60Hz at 1280x1024 should use this mode.

**High** This option defines a high scan rate monitor. Monitors that are capable of 74Hz Vertical refresh rate at 1280x1024 should use this mode.

### Mouse Type

This defines the type of mouse connected to the system on COM port 1. The Type of mouse connected to the system. This can be one of the following:

MouseSystems	Mouse systems 3 button mouse
Microsoft	Microsoft 2 button mouse
LogitechLogitech	Mouse
BusMouse	For BUS mice including Logitech versions.

### Font Paths

This list indicates the paths searched for fonts by the XServer. These paths are all local to the BXTerm. In network boot Bxterm's they will be situated in the BXTerm sub-directory "usr/X11R6/lib/X11/fonts". Generally it is not recommended to change these settings unless you want to add access to a font server. Adding access to a font server can be accomplished by adding a new path after the listed ones. The path should have the form:

<remote path>,tcp/<host name>:<port>

For example to allow access to the fonts in the directory "/usr/X11R6/lib/X11/fonts/misc" of the host "fred" which has the X fontserver running on port "7100" use the following entry:

/usr/X11R6/lib/X11/fonts/misc/,tcp/fred:7100

Note that the host has to have the X-Windows font server running for this to work.

### Raw Config Edit

This option brings up a text editor to edit the XF86Config file. This file contains the raw information as read by the X-Terminal. It can be used to change non-standard options that the Edit XConfig menu does not provide for. Note that error in this file can cause the XServer to fail to boot correctly. Only use this option if you have read the manual on the file XF86Config and know what you are doing. If the Xserver fails to run the BXTerm software will indicate an error and allow you to re-start the XServer with the default configuration. You can now re-edit the XF86Config file to sort out the problem.

### Reset Config

Resets the configuration to the factory defaults.

### Printer Configuration

This dialog box allows you to configure the printer on the Bxterm. The only option provided is to turn the printer on or off.

### Load Fonts

This dialog box provides the ability to load user defined fonts into the X-Terminals user font space. The dialog shows the current user fonts in a list down the left hand side. The buttons on the right allow you to delete fonts or append new fonts. The fonts have to be in either the BDF or PCF formats. The following commands are provided:

Delete font	Deletes selected font
Append from NFS host	Appends a list of fonts from and NFS host.
Append from floppy disk	Appends a list of fonts from a floppy disk.
Update font directory	Updates the fond directory information (mkfontdir). This is only necessary if you manually load fonts into a directory.

### Load Fonts from NFS hosts

This dialog box provides the ability to load user defined fonts into the X-Terminals user font space from a host supporting NFS. In order to do this the host must allow the BXTerm permission to mount the directory for NFS read access.

## STEPS

Place the fonts in either BDF or PCF format into the directory on the host machine.

Enter the host name or internet address and the directory into the dialog entry boxes.

Press the OK button to load the fonts

## DIALOG ENTRY BOXES

**Host**                    The Name or Internet Address of the host where the fonts reside. Note that it is only possible to use a host name if the nameserver option in the network configuration is set up correctly.

**Host directory**      The Directory on the host where the fonts reside.

## **Load Fonts from a Floppy Disk**

This dialog box provides the ability to load user defined fonts into the X-Terminals user font space from a floppy disk. The fonts must be stored on a floppy disk with a RAW tar file stored on it. The floppy should be formatted 1.44 MBytes with no file system on it.

## STEPS

Store the files, either BDF or PCF format, onto the floppy disk in RAW tar format.

```
tar -cf /dev/rfd0 *.bdf
```

Press the OK button to load the fonts

**NAME**

xdmhosts – BEAM Xchoose options file

**SYNOPSIS**

**xdmhosts**

**DESCRIPTION**

The file *xdmhosts* contains configuration information used by the BEAM Bxterm. It is read by *xchoose* after the X-Terminal has booted and contains a list of possible hosts for login together with the mode of login.

Each line of the file has the following format :-

[Name of Host]:[Mode of login] [Internet address of host]

[Mode of login] may take the values *xdm* or *telnet*

For example :-

beam:xdm 136.46.7.78

moonbeam:telnet 136.46.7.79

# Appendix A

## Terminology

### Boot server

This host is to be the Unix machine loaded with the Bxterm host kit Tape. It will be the host that the Bxterm's boot from and request certain run time information.

### Internet Address

An internet number is a number of form

[0-255].[0-255].[0-255].[0-255]

For example 137.44.4.64 is an internet number. It is used to identify host on a network. The Boot server internet address is the identifier for the designated machine.

### Host Name

Recalling machines by internet number can be tricky. They are more generally given 'meaningful' names such as bxterm1, etc. The name to internet address mappings is performed either by a nameserver or by a file.

### Network Mask

The network mask identifies a sub-set of the internet addresses. This sub-set comprises all the possible addresses a machine can contact without having to go through a gateway. Normally the network mask, masks the top 8, 16 or 24 bits of a hosts internet address. The network class A,B or C defines these types of network address.

Class	Netmask	Network Type
A	255.0.0.0	1 Byte Network, 3 Bytes Host Id
B	255.255.0.0	2 Byte Network, 2 Bytes Host Id
C	255.255.255.255.0	3 Byte Network, 1 Byte Host Id

### Gateway

A host machine on the local network providing access to other networks.

### Default Host

The name of the default host for login.

### Xdm Host

Network host providing X display management.

## Appendix B

### Hardware Compatibility

#### Motherboards

The Bxterm software operates on IBM PC compatible computers with either 386, 486 or Pentium processors. There is a minimum requirement for 8MByte of memory. To load the software (Disked versions) an unused partition with a minimum of 50MBytes of free space must be available on the hard disk.

#### Network Cards

##### SMC Elite 16, SMC Ultra and SMC EZ Ethernet Cards

The recommended configuration is:

IO ADDRESS	0x280
IRQ	5
RAM ADDRESS	0xD0000

Use the MSDOS SMC supplied ezsetup utility to reconfigure the card.

##### Novel NE2000 Ethernet cards

The recommended configuration is:

IO ADDRESS	0x300
IRQ	5

#### Graphics Cards

##### Genoa 8500 Series

The Bxterm operates with the GENOA 8500 series of Graphics cards. These cards are based upon the popular Cirrus Logic 542x series chipset.

##### ET4000 Cards

The Bxterm operates with GENOA 7900 ET4000 based graphics cards.

##### Mach32

The Bxterm operates with the ATI Mach32 graphics cards.

##### S3 Graphics Cards

The Bxterm has been tested with a range of S3 Graphics cards including those based on the S3 864 and Trio64 chipsets.

## Appendix C

### Interactive Unix V/386

#### Problems with Interactive V/386 and X-Terminals

Interactive Unix V/386 has two problems with X-Terminals:

**Xdm**        The xdm as supplied cannot support X-Terminals and the hosts display at the same time.

**Xterm**      The xterm as supplied will only work with capital characters unless the keyboard mappings are changed.

A optional disk supplied upon request **Xterm** includes new versions of these programs that cures these problems. To install the disk:

1. Login as root
2. Move the old versions of xdm and xterm to xdm\_1 and xterm\_1, both are in the directory /usr/bin/X11.
3. Cd to / and load the floppy using the following command:  
`cpio -icdamuBI /dev/rfd0`

when the the system is rebooted these changes will take effect.

## Appendix D

### Configuration System

With the inclusion of the Bootp protocol to gather network configuration information, the Bxterm configuration system has become a bit more complicated. This appendix lists the algorithm used to determine the network configuration.

#### Installation Floppy Disk

The sequence for determining the configuration is as follows:

1. The default configuration is loaded from the floppy disk. This configuration can be changed and saved onto the floppy disk.
2. If an XVI/Bxterm system exists on the hard disk, the configuration is updated from the configuration stored there.
3. The Bootp client tries to get the network configuration information from a bootp server. If it can it updates the configuration with this information. If the **T200** Bootp options is not specified the **UPDATE:** variable (Network update host) is set to the directory */opt/bxterm* on the **Bootp host**.
4. The final configuration is used for installation and stored onto the hard-disk.

#### Normal System boot

The sequence for a normal system boot is as follows:

1. The configuration as stored on the hard-disk is loaded.
2. If the **BOOTP** option is set on, the bootp server is contacted for information. If it responds then the configuration is updated. If the **T200** option is not set, then the current value is used.
3. If the **UPDATE** option (Network Update host and directory) is set then an NFS mount is attempted to check the versions of software present on the update host. This will time out if not accessible, with warnings. If a new version is found then the user is asked if they wish to update. This will timeout if no response is received from the user.