courBR12.snf	-adobe-courier-bold-r-normal--12-120-75-75-m-70-iso8859-1
charBI10.snf	-bitstream-charter-bold-i-normal--10-100-75-75-p-72-iso8859-1
charBI12.snf	-bitstream-charter-bold-i-normal--12-120-75-75-p-74-iso8859-1

1.9.6.4 Making a *fonts.alias* File

Another commonly used font file is optional, but is sometimes used to substitute different names for specific fonts. This is the *fonts.alias* file. The ViewStation looks for this file and uses it if it is present.

The format of the *font.alias* text file is:

<*alias*> <*white space*> <*fontname*>
<alias> <white space> <fontname>
<alias> <white space> <fontname>

For example, a *fonts.alias* file might be:

```
*-courier-bold-r-normal-*-*-120-*    9x15
*-courier-bold-i-normal-*-*-120-*    9x15
FILE_NAMES_ALIASES
```

This means that if a client requests the Courier Bold Italic 12 font, the ViewStation checks the *fonts.alias* file and substitutes the 9x15 font.

The string FILE_NAMES_ALIASES is a special keyword which creates an alias for each font file listed. The alias is formed by deleting the file extension from the font file name. Thus the alias for 6x10.snf is just 6x10, which can be used when the client calls that font, as in **xterm -fn 6x10**.

Create this file with a text editor, such as *vi*, and save it as “fonts.alias” in the HDS font directory. Be careful with spelling in this file. If a name is misspelled, the font won't be aliased correctly. If the keyword FILE_NAMES_ALIASES is misspelled, this function won't operate.

path (a good idea), you can do it with the xset command as follows (the command **xset +fp** puts the new directory at the beginning of the search path and **xset fp+** puts it at the end):

```
xset fp+ /HDS\$\$ROM/
```

At this point, your **xset** query should show the correct fontpaths, and your **xlsfonts** query should list all the fonts you have available. You can test the operation of your loaded fonts by running an xterm program specifying one of the particular fonts in your fontpath. Use a command line such as:

```
xterm -fn timB18
```

to load the 18 point times bold proportional font. This xterm will look strange because of the proportional font, but the font appearance is distinctive so you'll know the font loaded correctly. If the font doesn't load, a default will often be selected and an error line appears saying the font couldn't be found.

If you specify a font by its entire font name (not its file name), you can use wildcard characters in the name, but you must use single quotation marks around the font name, such as:

```
xterm -fn '*times-bold-*-180-*'
```

The wildcard characters are special UNIX characters and the entire name must be quotation marks or the wildcards each preceded by a backslash \, such as:

```
xterm -fn \*times-bold-\*-180-\*
```

1.9.6.3 Making a *fonts.dir* File

A *fonts.dir* file is essential for the correct operation of X font handling. This file MUST be present in your fontpath directory. It is a kind of database for fonts that the ViewStation, and all X servers, use to locate fonts. You must create a *fonts.dir* file after all your conversions are complete and the unnecessary .bdf files have been removed. This file is necessary for any type of font loading. Make certain you are in the correct HDS .snf file directory. If a *fonts.dir* file exists, delete it; then just issue this command:

hdsmkdir or **mkfontdir**

This command automatically creates a *fonts.dir* file. Verify that it appears in the directory by using the **ls** command. The format of the *fonts.dir* text file is:

```
< number of fonts in the file >
<filename without path>   <white space>   <fontname>
<filename without path>   <white space>   <fontname>
<filename without path>   <white space>   <fontname>
```

For example, a typical (but untypically short) *fonts.dir* file would be:

```
4
courBO10.snf                    -adobe-courier-bold-o-normal--10-100-
75-75-m-6-iso8859-1
```

server will try each host which matches the local mount directory until it finds one that responds. The Local Mount Directory specified must be the same for each host, and only one font path using that mount point is entered in Setup Mode.

For example, on the Remote host menu:

Remote Host 1:
Host Name: hdssun3
Network address: 128.91.3.1
Local Mount Directory: /fonts
Remote Mount Directory: /usr/lib/X11/fonts/hds-fx/fonts

Remote Host 2:
Host Name: hdssun6
Network address: 128.92.7.5
Local Mount Directory: /fonts
Remote Mount Directory: /home/HDS/hds-fx/fonts

and so on.

If the font path entered in Setup Mode is "/fonts/75dpi,/fonts/misc,/fonts/100dpi", then Remote host 1 will be contacted first for all fonts, with Remote host 2 active as a backup in case host 1 is not available.

1.9.6.2 Verifying the Fontpath Setting

You can use the xset command to set the fontpath. For instance,

```
xset fp= /usr/lib/X11/hds-fx/fonts/100dpi/
```

The command to update the fontpath setting is:

```
xset fp rehash
```

The HDS ViewStation automatically updates the fontpath when you Apply Changes in Setup Mode (note the message at the bottom of the Setup Mode window). You will probably also want to save the fontpath setting in the ViewStation’s NVM. If you have changed the fontpath, you should verify that the fontpath is correctly registered or save it in NVM and restart the ViewStation to be certain it is correctly read.

You can check the fontpath with the **xset** query command:

```
xset q
```

which shows many of your current server settings, including the fontpath.

You can also use the **xlsfonts** command to show the available fonts in your fontpath. This is useful because it shows the valid aliases as well as normal fonts.

If you have set the fontpath in Setup Mode, you will notice that the fontpath includes HDS$$ROM in addition to your designated directory. This indicates that the ROM fonts are in the search path. If you wish to append them to your fontmetric. On SUN systems, scalable fonts are converted to various size font files with the **makeafb** utility program, and these *.afb* files are converted to *.bdf* files with the **convertfont** utility. Other systems may use different formats and/or different utility conversion programs.

All of these font format types may be converted to HDS .snf format by using the HDS font utility programs, specifically, the **hdsfc960** font compiler. This compiler takes font files of any format (even if they can be read only by the particular server they belong to), reads them with general X routines, and creates HDS format .snf files.

1.9.6 Loading Fonts to the ViewStation

There are a few simple requirements for using fonts. The ViewStation can read .pcf, .bdf, or .snf font formats directly. If you are using a font server, the server takes care of differences in server format, so you don't have to have .snf fonts for the ViewStation.

- Fonts must have the correct format (.pcf, .bdf, or .snf). This was discussed above.
- The fontpath must be correctly given to the ViewStation so it can locate the fonts. If you're using a font server, its path must be included.
- There must be a fonts.dir directory.
- There should be a fonts.alias directory (optional).

1.9.6.1 Setting the Fontpath

The fontpath is entered automatically if you use the automatic HDSware configuration feature (see Section 1.3.4) of the HDS ViewStation. You can also enter the fontpath in Setup Mode on the User Preferences menu (see Section 1.5.6.5).

A typical path might be:

```
/FXUtilities/hds-fx/fonts/75dpi/,/FXUtilities/hds-fx/fonts/misc
```

where FXUtilities is the HDSware host and path.

If you have a font server, your path might be:

```
tcp/hostA:7000
```

You can use multiple hosts and multiple directories separated by commas. If one host or directory is not reachable, the ViewStation still uses the remaining ones. If the same .snf fonts are located in more than one path, or on more than one host, only the first path correctly reached will be used.

Backup Font Hosts

The font server has an alternate font server list as part of its configuration. This is discussed in Section 1.9.3.2.

If you use a regular fontpath, you can specify that a font host act as a backup, and there is a little trick to make this happen. You specify the same "local mount directory" for each host, then specify the remote directory for each host (on the Remote hosts menu in Setup Mode), so the resulting local font paths are the same. The vided for monitors with different screen resolutions. (See Section 1.1 for dot resolutions of various FX monitors.) The appearance of a font on the screen is best when the dots per inch resolution on the screen is as closely matched as possible to the number of dots provided for each character in the font file.

(This can become a rather complicated question for users concerned about exact character size. For instance, a 24 point font from the 75dpi directory displayed on a monitor whose resolution is actually 85 dpi will be slightly smaller than 24 points. [The calculation goes like this: Since a point is about 1/72nd of an inch, a 24 point character from the 75dpi directory is about 24 dots high. This means that 24 dots displayed on 85 dpi results in a smaller character - 24/85ths compared with 24/75ths of an inch.] On the other hand, the same 24 point font from the 100dpi directory will be larger than 24 points when displayed on the same monitor [24/85ths compared to 24/100ths of an inch]. Since most applications where this is important, such as WYSIWYG publishing programs, take these factors into consideration with internal mechanisms for their screen displays and printing, most users won't need to be concerned about it. If this is an important consideration, put the preferred font directory, 75dpi or 100dpi, first in your search path. This will insure that when a particular font is requested [if it is available in both directories] it will be loaded from that directory.)

The HDS ViewStation uses either 75dpi or 100dpi font files, depending on the monitor used. Some application programs specify specific dpi values in their font requests and may require loading both 100dpi and 75dpi font directories. The correct font size is calculated by measuring the horizontal display width in inches, then dividing it into the horizontal resolution of the screen to give pixels/inch (refer to Section 1.1 for these values). Then choose the font resolution closest to that value, 75 or 100 dpi. This information is also provided by the X command **xdpyinfo**.

If your .bdf fonts appear as files with a .Z extension, they are compressed files (all files from the MIT distribution are compressed). You can convert these files easily with the uncompress command:

```
uncompress filename.bdf.Z
```

Or, using a wildcard character:

```
uncompress *.bdf.Z
```

The new files appear with the .bdf extension only. They then may be compiled into .snf font files.

Other systems, such as OPEN LOOK (usually *.ff* or *.f3b* font format), Hewlett Packard (in .scf format), or DECwindows (in *.dwf* format), have their own sets of precompiled fonts and font directories necessary for their operation. Some of these fonts may be in a scalable format (which means that they are general formulas for fonts and individual font files, in the desired sizes, must be created before they can be used); scalable fonts often have the *.f3b* extension on their file name, and their extended font name has 0-0's in the size and spacing fields. Other common font formats include *.afb* and *.afm*, which are Adobe formats, bitmap and

1.9.5.3 hdsmkdir Font Utility Program

Creating a *fonts.dir* file with **hdsmkdir** is also a simple operation. It works like the **mkfontdir** command, but it works on any machine architecture and accepts more than one directory name.

1.9.5.4 getbdf Font Utility Program

The **getbdf** program creates a .bdf font file from a file of different format. It reads the font data using various X tools (xlib), and reconstructs the original .bdf file. This utility is especially useful for systems that have only precompiled font files.

1.9.5.5 hdsfc960 Font Utility Program

The **hdsfc960** utility program is more general. It creates HDS .snf font files from any font file type, including precompiled fonts. It uses X tools (xlib) to read the font from the host server display, then creates a .bdf file from that font data using **getbdf**, then creates the .snf file using **bdfto960**, and finally creates a *fonts.dir* file using **hdsmkdir**.

To use **hdsfc960**, you must first create a directory for the HDS .snf fonts. It can be at any location and can have any name, but it must be empty. The **hdsfc960** program converts all the font files in the current fontpath, so you should be careful to limit the fontpath on the target server to those fonts you want to recompile. (The best way to do this is to create a separate directory containing only the fonts you want to recompile, then restrict the fontpath of the host target display to that directory.) The command line is:

```
hdsfc960 -d <IP address of target display:display> <destination directory>
```

or, for example,

```
hdsfc960 -d 128.91.6.1:0 /usr/XllR5/HDS/fonts/100dpi
```

You should enter the target server IP address (the target server is a display - such as the host's console - that can display the fonts) explicitly so you don't convert any fonts you don't want to. The **hdsfc960** program will convert all fonts in the target server's fontpath, so you should use **xlsfonts** beforehand to make sure the font list is correct.

Note that **hdsfc960** uses the other HDS font utility programs, so they must be available and have the correct execute privileges set.

1.9.5.6 Handling .bdf Fonts and Fonts in other Formats

There are large directories of .bdf font files supplied with the standard MIT X distribution. You may not want to use all of these fonts or create .snf files for all of them. This choice depends on the X clients and applications you want to use. All of these fonts must be compiled into .snf format to be used on the ViewStation (or use the pre-compiled HDS .snf fonts from the HDSware tape).

Font Sizes

Other important font considerations include choosing the correct 75dpi (dpi = dots per inch) or 100dpi files for your HDS ViewStation. These sets of files are pro-

These utility programs are available in the HDSware distribution in the directories *./hds-fx/utils*. There is an */execs* directory with executable files for a number of host systems. There is also a */src* directory with source code and sample files for compiling these utility programs for other hosts.

1.9.5.2 bdfto960 Font Utility Program

If you have .bdf files you want to convert, use **bdfto960**. The command line arguments are set by default to the HDS ViewStation format. This compiler is similar to the MIT **bdftosnf** compiler except it works on any type of machine architecture. You can use it in the same way as the MIT compiler.

For instance, if you want to manually convert some .bdf fonts, use this procedure. The HDS **bdfto960** utility has the command line arguments set automatically, but you must enter them manually with the MIT **bdftosnf** command.

Converting .bdf files to .snf files with the MIT **bdftosnf** command is done with the command line specifying the arguments for the HDS ViewStation architecture:

```
bdftosnf -p4 -u1 -L -l  file.bdf > file.snf
```

where:

-p4 sets glyph padding to 4

-u1 sets scanline padding to 1

-L sets Least Significant byte first

-l sets Least Significant bit first

Converting the large numbers of fonts can sometimes be a lengthy process. This little program will help you handle the conversion process by processing all the files in a loop. These same programs can also be used with **bdfto960**, but it is not necessary to enter the arguments on the command line.

For the C Shell: /bin/csh

```
unix% foreach i (*.bdf)
        echo $i
        bdftosnf -p4 -u1 -L -l $i > $i:r.snf
    end
```

For the Bourne Shell: /bin/sh

```
unix$ for i in *.bdf
    {
        echo $i
        bdftosnf -p4 -u1 -L -l $i > `basename $i .bdf`.snf
    }
```

(Note that ` is an accent grave; be careful with spaces in these commands) files (refer to the discussion below). If you don't have .bdf files but only have .snf files, you can use the **hdsfc960** compiler, described in Section 1.3.6.

1.9.4.3 Font File Formats - .bdf and .snf

Font files contained in the MIT X distribution are in .bdf format (Bitmap Distribution Format, a generic, human-readable format). Other systems, such as DECwindows or Open Look, provide fonts in other formats, such as .dwf (DEC Windows Format) which are pre-compiled for the particular machine architecture they use. These fonts must be recompiled into a special .snf format (Server Natural Format) specifically for the HDS ViewStation (or the .snf format for the machine they are running on). Each server type has different .snf format requirements, based on its memory configuration and processor type, bit and byte order, etc. Other servers will use different .snf files, and you must create specific ones for the HDS ViewStation, as shown below.

The first step is to create a special directory for these font files. Then the .bdf fonts you need should be copied into the directory and converted to the .snf format for the ViewStation.

1.9.5 Using HDS Font Utilities

There are a number of font utility programs available from HDS. These utilities include a general **hdsfc960** utility for font management and other more specific programs. These programs are available in executable form for many types of systems, or as source code for all systems. The **hdsfc960** font utilities are available from any HDS office.

The HDS font utility programs create .snf files in HDS ViewStation format from font files of any format. They read the font data from a remote host server display (such as the system console) using normal X protocols and reconstruct the .bdf file, then use their own **bdfto960** compiler to make the .snf font files. These font utility programs can use either generic .bdf files or precompiled font files, such as .dwf or .ps. They can also run on any kind of host architecture and create properly formatted files.

1.9.5.1 HDS Font Utilities

HDS has prepared a set of font utility programs to make it easier for HDS ViewStation users to compile and manage font files. These programs are included on the HDSware tape, along with X11R4 fonts which have been precompiled for use with the HDS ViewStation. This tape is available from any HDS office.

There are several separate font utility programs:

hdsfc960 - The general program to make .snf files from any font source file

bdfto960 - The program to make HDS .snf files from .bdf files

hdsmkdir - The program to create a *fonts.dir* file for a font directory

getbdf - The program to create .bdf files from any font source

220 KB of memory to load fonts (12 fonts) with conventional font loading, but uses no memory with a font server. This case illustrates the savings possible with font server usage. Depending on your use of fonts, this could be a benefit to your memory consumption.

1.9.3.3 Using Alternate Font Servers

If you want to have a backup font server, you can specify alternate font servers in the font server's configuration file (see the font server man pages for syntax). A typical entry would be:

```
alternate-servers=hostA:7000,hostB:7000,hostC:7000
```

Note however that these lists replace each other, so if hostA is unavailable, the alternate list used is from hostB and will not retry hostA unless it appears in the new list. It is therefore recommended that each host font server show all of the other font servers in their lists.

1.9.4 HDS ViewStation Font File Formats

The ViewStation can use .bdf, .pcf, or .snf font files directly. If any of these font formats are available to the ViewStation, it can use them. The .pcf fonts are usually available with an X11R5 font server.

A complete set of compressed .snf font files, in HDS ViewStation format, is available directly from HDS. You may contact your sales representative or distributor for more information about these pre-compiled fonts. If you use these fonts, you can load them onto a host and then set the fontpath automatically by using the HDSware host described above, or you can set the fontpath manually as described below.

1.9.4.1 ViewStation Compressed Font Files

The ViewStation uses font files in either compressed or uncompressed form. The font files are normally supplied as compressed files (with the .Z file extension) on the HDSware tape. You do not need to uncompress these fonts to use them on the ViewStation.

If you are compiling your own fonts, you can use fonts in an uncompressed form.

If you do need to compress the fonts, use 12 bit compression rather than 16 bit compression. The normal 16 bit compression uses a larger, less efficient table, so 12 bit compression is recommended. The command to use 12 bit compression is:

```
compress –b 12
```

1.9.4.2 Making Special .snf Font Files

If you have special application fonts or your own custom fonts (typically in .bdf format), you must use the **bdfto960** compiler supplied by HDS to compile them. If you have .bdf fonts the **bdfto960** compiler is specifically designed to create .snf fonts for the HDS ViewStation architecture and can be used in all situations. You can also use the **bdftosnf** compiler supplied with the MIT X distribution, but this does not work correctly in all situations and may not produce workable .snf font

(File Transfer Protocol), or NFS (Network File System) to locate and load the fonts dynamically on demand. You can make this protocol selection for each remote host from the Remote Hosts menu in Setup Mode.

1.9.2 Automatic Fontpath Configuration

The HDS ViewStation has features designed to make installation of fonts and fontpaths very simple. The fonts and font utility programs, along with other important files, are contained on the HDSware tape. When you load this tape onto one of your host systems, all that remains for you to do is enter that host address and directory location on the Setup Mode Main Menu as the HDSware host and path.

The ViewStation takes this information and prepends it to the directory structure of the tape and uses that information to locate the fonts. The fontpaths, the server code files, configuration files, etc. are all located automatically, so you don’t have to enter them manually. The fonts distributed with the HDSware tape are in the correct .snf format and are ready to use.

The automatic fontpath configuration does not include font servers (which is a host-based service) but lists only the X11R5 font directories in the HDSware distribution.

1.9.3 Font Server Operations

The font server is a host-based X client that compiles fonts into the correct server format on demand. This client is usually started by the system administrator from boot files like */etc/rc.local*. You should be sure the font server is running if you intend to use it for your font access.

1.9.3.1 Putting the Font Server in the Font Path

The ViewStation’s font path has to have an entry for the font server. This is typically a simple entry which specifies the hostname and the port number:

```
     tcp/<hostname>:<port>
or   tcp/hdssun1:7000
```

The default port number is usually 7000, but it can be assigned to other ports by your system administrator.

This entry should be included in the ViewStation’s font path in the normal way. For example, a fontpath might be:

```
tcp/hdssun1:7000,/home/HDSware/hds-fx/fonts/Xol
```

1.9.3.2 Font Server Benefits

With normal font loading, a font file is transferred from the host’s font directory into the ViewStation’s memory. But with font server operation the ViewStation opens a connection to the font server on the host computer and the fonts are not actually loaded or cached by the ViewStation.

This can result in some saving of memory space on the ViewStation. For instance, a ViewStation starting up with Xdm and then starting HPvue with HPterm uses

HDS ViewStation:
Font Handling

This chapter discusses the use of fonts on the ViewStation, including loading processes, font utility programs, and handling custom fonts.

Section 1.0 - HDS ViewStation

1.9 Font Handling

Font handling has changed with the release of X11R5. Under X11R4, each server (X terminal or workstation) had to use a set of .snf fonts specific to the server architecture. Separate directories of .snf fonts were maintained for each server type. An snf font file was loaded on request of the server.

A font server program was introduced with X11R5 that used portable, scalable fonts that were converted by the font server to a given server format on demand. This is a more economical method of font management.

1.9.1 Fonts in the X Window System

Many clients in the X Window System use special display fonts for their screen presentation. The fonts may be large or small, use bold or italic forms, or use different typefaces. To accommodate the needs of all the X clients, the X Window System provides a number of typefonts as part of its standard distribution. An X terminal must be able to find and use these fonts to handle X clients successfully.

The X Window System fonts offer different type styles, different sizes of characters, and different screen font resolutions for the characters (for instance, 75 dpi or 100 dpi). In addition, there may be different character sets involved, including multinational and non-Roman alphabets, such as Kanji and Katakana. Further, the X Window System supports a number of clients that use fonts from different sources, such as DEC proprietary fonts, Open Look fonts, or Postscript fonts.

These fonts are made available to the server from a font server or from a remote host. The HDS ViewStation uses either TFTP (Trivial File Transfer Protocol), FTP