Update Notice:

DG/UX™ X Window System for AViiON® Systems

Release 5.4, Update 1

February 1992

Part number 078-600080-00

This release notice applies to the following models:

P001

Update Notice:
DG/UX™ X Window System for AViiON® Systems
Release 5.4, Update 1

078-600080-00

February 1992

Contents

1 Introduction

This update notice describes update 1 for the DG/UX ™ Window System Revision 5.4 release and its installation. This update will be referred to as 5.4.1 throughout this document.

This update notice supplements or supplants information found in the *DG/UX Window System Revision 5.4 Release Notice*. In addition, this notice includes information not currently available in the product manuals (e.g., information developed after the current manuals were printed, or corrections to current manuals).

Between releases, Data General may issue updates to the product. An update is essentially a partial release. This mechanism reduces the time needed to fix problems by providing a level of correction short of releasing the complete product. Each update of a product supersedes the previous update.

You may print additional copies of this update notice after you have installed the DG/UX Window System. A copy suitable for line printers can be found in the file **/usr/opt/X11/release/x11_5.4.1.un**. In the event of (actual text) differences between the printed copy of this notice and the online lineprinter version, the printed copy takes precedence.

2 Product Description

DG/UX™ Window System is a combination of the X Window System™ Release 4, Motif™ Version 1.1.3, and Looking Glass® Version 2.2. The X Window System consists of an X server, a set of X clients, programming libraries, and on-line documentation. The X server manages a raster display and related input devices. X clients are applications which communicate with the X server to perform terminal emulation, window management, and other necessary functions. The subroutine libraries allow programmers to implement applications of their own.

Motif consists of the mwm window manager, a User Interface Language (uil) compiler, and subroutine libraries. Motif is a derivative product to Open Software Foundation, Inc's OSF/Motif™.

Looking Glass®, a desktop manager, provides an icon-based interface to DG/UX™. Users can move and edit files, start programs, and view directories by using the mouse to manipulate icons.

3 Environment

3.1 Hardware

The DG/UX Window System is based on a client/server model. This model allows the DG/UX Window System to be run in two distinct environments, depending on whether both the client and server run on the same machine, or on different machines that are connected by a Local Area Network (LAN) using TCP/IP. In the first environment, where the client and the server are running on the same machine, they communicate using Unix sockets or shared memory. In this situation the minimum configuration for an AViiON workstation is a keyboard, mouse, raster display, disk and tape or fileserver.

In the second environment, the server and clients execute on different machines connected by a Local Area Network (LAN). The machine running the server must have, at a minimum, the standard AViiON workstation configuration. The machine running the client can be any CPU connected to the AViiON workstation by LAN.

3.2 Software

In both operating environments, revision 5.4.1 of the DG/UX operating system is a prerequisite. Consult the DG/UX release and update notices to determine its environmental requirements.

DG/UX Window System Release 5.4.1 fully interoperates with previous releases of the DG/UX Window System product.

4 Enhancements and Changes

4.1 Enhancements

4.1.1 xman

The widget which displays the manpage text is now a subclass of the Athena Text widget, and inherits all the capabilities of that widget class. Most notably is the ability to search for strings within the manpage text, and cut text to be pasted into another window. Also the menu system has been reorganized.

4.1.2 xdgmail

The **xdgmail** client application provides easy access to the Unix mail facility. Users can send and receive Unix mail through **xdgmail**'s OSF Motif compliant interface. Functions include support for composing messages, viewing messages, saving messages to files, printing messages, searching messages in a mailbox, replying to messages, forwarding messages, and organizing messages into convenient mail folders. **xdgmail** can be easily customized using the Motif resources.

4.1.3   mterm

The follow enhancements have been made to **mterm**:

- Added support for 7/8 bit being changed by the application.

- The logging menus have been consolidated into one menu. This reduces the need to go to multiple logging menus to manage the logging features.

- The user can now set the duration of the BREAK key.

- Added support for more AViiON keyboards.

- Turn off compose menu button if in 7 bit mode.

- A file selection box has been added to the menu system. It allows for easy filename browsing/selection.

- The cursor changes from a block to a box when the window loses focus. It is changed back to a block when the window regains focus.

- Support has been added for 7 bit international character sets.

- The user can define the pointer.

- Added select code for dragging a line or a word at a time.

- Draw cursor as a double line box when over a reverse video character.

- Updated warnings section of manpage. See **mterm(1X)** for complete information.

4.2   Changes

4.2.1   xterm utmp entries

Multiple usernames can correspond to a single uid number. In release 5.4, the utmp entry created by **xterm** would contain the first name in the passwd database matching the current uid number. **xterm** has been changed to use the username entered when logging into the system.

4.2.2   xterm Home and End keys

In release 5.4 **xterm** would incorrectly interpret the Home and End keys. **xterm** has been changed to not generate any escape sequence for these keys, because the terminal **xterm** is emulating, a VT102, does not have either of the keys. This default behavior can be overridden with the "translations" resource. See the **xterm(1X)** man page for more details.

4.2.3   **mwm protocol**

mwm now handles the WM_SAVE_YOURSELF protocol independently of the WM_DELETE_WINDOW protocol. A client expressing interest in both protocols will be notified for each (WM_DELETE_WINDOW, then WM_SAVE_YOURSELF) when the user invokes f.kill on a window. A client expressing interest in neither protocol will have its connection to the X server terminated. After mwm sends a WM_SAVE_YOURSELF client-message, it sets a timeout; the value is specified by mwm's quitTimeout resource. The client's connection to the server will be terminated when the timeout expires or when the client has updated its WM_COMMAND property.

4.2.4   **X11 directories**

In release 5.4 the directories **/usr/bin/X11**, **/usr/lib/X11** and **/usr/include/X11** were symbolic links to directories in the **/usr/opt/X11** filesystem. This caused problems when installing third party software for the following reasons:

● When **/usr/opt/X11** is remote mounted, installation of third party X11 software will fail. Software installation is performed by the user "root". Unfortunately across an NFS mount the user "root" is translated to the user "nobody" on the remote system. And the user "nobody" does not have the write access.

● Files added to these directories are actually added to the **/usr/opt/X11** filesystem instead of the **/usr** filesystem. This will make disk space instructions incorrect. For example, a third party package may instruct the user to have a certain amount of free space in the **/usr** filesystem, while actually the free space is required in **/usr/opt/X11**.

● Relative symbolic links (links starting with "..") added to these directories could be incorrect. For example symbolic links created in the directory "**/usr/bin/X11**" are actually created in "**/usr/opt/X11/bin**". The problem is that "**/usr/opt/X11/bin**" is 4 directory levels below root ("/"), instead of 3 as **/usr/bin/X11** implies. The *Porting and Developing Applications on the DG/UX*™ *Systems* (069-701059), referred to as the DG/UX Porting Guide in the rest of the document, recommends using absolute symbolic links (links starting with "/") in these directories, to avoid this problem.

In release 5.4.1 these directories are real directories in the **/usr** filesystem, and the individual files in the directories are linked, instead of the directories themselves. The following is the list directories that are created in the **/usr** filesystem:

```
/usr/bin/X11
/usr/include/X11
/usr/include/X11/bitmaps
/usr/lib/X11
/usr/lib/X11/app-defaults
/usr/lib/X11/fonts
/usr/lib/X11/fonts/100dpi
/usr/lib/X11/fonts/75dpi
```

```
/usr/lib/X11/fonts/misc
```

The setup script "**/usr/sbin/setup.d/usr/X11__0.X11.do**" creates the directories listed above, and creates symbolic links each of the files contained in corresponding **/usr/opt/X11** directory (e.g., **/usr/opt/X11/bin** for **/usr/bin/X11**).

There are 3 methods third party software can use for installation (traditional UNIX, DG/UX Porting Guide, and other). The following paragraphs will explain how this change effects each of the methods. The package "**foo**" installing the file "**bar**" into the directory **/usr/bin/X11** will be used in the examples.

The traditional UNIX method views the **/usr** filesystem as one large filesystem containing all the software on a system. This method is based on earlier versions of UNIX which did not have symbolic links, and had limited support for Logical Disk Units (LDU). The files to be installed are simply copied, moved or tar'd into the appropriate directory.

```
cp/mv/tar -> /usr/bin/X11/foo
```

This method would fail on a release 5.4 system which remote mounted the **/usr/opt/X11** filesystem. Furthermore, this method assumes the files are being added to the **/usr** filesystem, so installation instructions may incorrectly describe disk space usage. In release 5.4.1, remote mounting and disk space usage are not issues, because the X11 directories, as this method assumes, are real directories in the **/usr** filesystem.

The method described in the DG/UX Porting Guide installs the files of an optional package in the opt sub-directory of a filesystem, such as **/usr/opt/foo** for the package "**foo**". Symbolic links are then added in "public" directories (e.g., **/usr/bin/X11**) which point to files in the opt sub-directories. The DG/UX Porting Guide recommends absolute symbolic links be used in the X11 directories. The DG/UX Porting Guide goes on to recommend these links be added as absolute symbolic links for the X11 directories. For example, the following command should be used to link the file "**bar**" in the package "**foo**":

```
ln -s /usr/opt/foo/bar /usr/bin/X11/bar
```

The X11 directory change in release 5.4.1 will allow the use of relative symbolic links, although, absolute symbolic links are still recommended, for downward compatibility with 5.4 systems. For example, the following 3 ln commands will work on a release 5.4.1 system, while only the last will work on release 5.4:

```
ln -s ../../opt/foo/bar /usr/bin/X11/bar
ln -s ../../../usr/opt/foo/bar /usr/bin/X11/bar
ln -s ../../../../usr/opt/foo/bar /usr/bin/X11/bar
```

Also a symbolic link in the following form will only work on a release 5.4 system:

```
ln -s ../../../opt/foo/bar /usr/bin/X11/bar
```

The third method of installation is directly accessing the **/usr/opt/X11** filesystem, as in the following examples:

```
cp/mv/tar -> /usr/opt/X11/bin/bar
ln -s /usr/opt/foo/bar /usr/opt/X11/bin/bar
ln -s ../../foo/bar /usr/opt/X11/bin/bar
ln -s ../../../opt/foo/bar /usr/opt/X11/bin/bar
ln -s ../../../../usr/opt/foo/bar /usr/opt/X11/bin/bar
```

An optional package should NEVER write in the installation area of another package. The **/usr/opt/X11** filesystem is only intended to segregate the X11 package files from the base DG/UX system. If a third party software package does install its files in this manner, the setup script **/usr/sbin/setup.d/usr/X11__0.X11.do[ne]** can be re-run to fix this situation. This script will create symbolic links for all the new files. Since this script is automatically run when a system is upgraded to release 5.4.1, it only needs to be re-run when installing a package, which loads directly into **/usr/opt/X11**, after a system has been upgraded to release 5.4.1.

4.2.5   **Software Trouble Reports Resolved**

Following is a list of Software Trouble Reports that have been resolved in DG/UX Release 5.4, Update 1. The problem descriptions given here are listed exactly as described in the original report filed by customers. Please refer to the DG/UX Monthly Newsletter for additional details.

- AUSC-000007565-0
  Unable to enter the BREAK sequence from an MTERM session running on an Aviion workstation. User gets the error TIOCSBRK: Invalid argument, returned to the session that invoked the MTERM session.

- JAPN-000000594-0
  Keycode range maximum is wrong after loggin on twice under xdm.

- NASC-000010093-0
  4.32 of xterm outputs incorrect key codes for the HOME and END key on AV300 workstations.

- NASC-000010441-0
  The X11 xterm client returns escape sequences that are different than what is defined in the terminfo database for the home and end keys.

- NASC-000010525-0
  Motif application aborts with memory fault - core dumped within 1 to 20 minutes after being started.

- NASC-000010595-0
  When logged in as user sysadm, the passwd command fails to set sysadm's password in all cases. If used without arguments (ie. password<CR>), the new password is applied to user root.

- NASC-000010995-0
  User's that use ftp and log in as xdm have root access to the system which results in a security leak.

- NASC-000011018-0
  Mterm does not function properly if -baud is set to 19200. Port tty00 is set to 9600 baud.

- NASC-000011257-0
  mterm -async -baud x defaults to 9600 regardless of the baud setting.

- RTP0-000013807-0
  Display of dashed lines on AViiON X Server is incorrect. A test program is attached.

- UKSC-000049326-0
  Per Manual: ic_env file can be 'sourced in' using dot command in .profile. When logging in using xdm a shell is executed using .Xsession as an argument. As a result ic_env aborts as it checks that no argument is passed to it.

5   Notes and Warnings

5.1   Notes

5.1.1   Editread Restriction

If **editread** is turned on when an **xterm** is started in the background (e.g., "**xterm &**"), the **stty** settings picked up by the background **xterm** could be those before **editread** sets them up or those after. **xterm** reads the **stty** settings of the parent tty (**/dev/tty**) to establish the settings it will use. **editread**, however, sets up the **stty** settings during the re-initialization after the background **xterm** is launched. It is, essentially, a race as to whether the **stty** settings are established by **xterm** or **editread**.

This condition appears often in the **csh**, but rarely in the **sh**. The workaround is to have a **.cshrc** file that resets your **stty** settings in **xterm** after the C shell comes up or to turn **editread** off before the launch. If you turn off editread before the launch, you cannot turn it back on.

5.1.2   **AVX30**

The AVX30 Release Notice instructs the system administrator create a symbolic link in the **/usr/opt/X11/lib** directory. Due to the change in the public X11 directory structure explained above, the command to create this symbolic link has changed. If you are installing X-terminals for the first time, use the following command:

```
ln -s /usr/opt/X11/xtd /usr/lib/X11/ncd
```

If the symbolic link had been created prior to upgrading the system to 5.4.1 using the AVX30 Release Notice instructions, then no action is necessary. The link will automatically be converted during the 5.4.1 package setup.

5.1.3   **X11 applications**

When loading an application (which loads files directly into **/usr/opt/X11**) onto a DG/UX 5.4.1 system, the X11 setup script may need to be re-run using the following commands:

```
/usr/sbin/setup.d/usr/X11__0.X11.done / /usr
```

For more information see the Changes section titled "X11 directories" above.

5.2   **Warnings**

5.2.1   **mwm resources**

The multiClick resource cannot be specified in a resource file because there is no converter available for it.

mwm now requires at least one key binding to be present in a "Keys" definition in the **.mwmrc** file, or it will default to using the built-in key bindings.

5.2.2   **Motif resource manager**

When using the MrmFetchWidget call, contrary to the man page, the user must specify the address of an already defined structure MrmType for the last parameter (class) as opposed to just a pointer. Failure to do so can possibly result in a core dump of the calling application.

6   **Documentation**

6.1   **Titles**

For a complete list of titles, see **/usr/release/doc_guide**.

6.2   **Changes**

A new manual page for **xdgmail(1X)** has been added.

7   Software Distribution

7.1   Media

The DG/UX Window System Release 5.4, Update 1 is available on two Data General product tapes. They are:

- Model Number P001, the DG/UX Operating System with X Windows package

The product media notice that came with your tape will identify the product and provide tape part numbers.

7.2   Organization

Since the DG/UX Window System product is bundled with other products, tape layout cannot be provided here. If it is desired, the **sysadm** command **software -> package -> list** can be used to display the exact layout of any release tape (prior to loading any files).

7.3   Files

For a complete list of files delivered with this update to the DG/UX Window System Package, see the following files under **/usr/opt/X11/release** after the system is loaded (associated package in brackets):

```
X11_5.4.1.fl          [X11]
X11.man_5.4.1.fl      [X11.man]
X11.sde_5.4.1.fl      [X11.sde]
X11.lg_2.2.fl         [X11.lg]
```

8   Installation Instructions

You must load release 5.4, Update 1 of the DG/UX Window System over a system loaded with release 5.4 of the DG/UX Window System. You may load DG/UX Window System 5.4 and 5.4, Update 1 in the same system installation session.

9   Preparing a Software Trouble Report (STR)

Report any particular activity or program running on the system that seems to cause the problem. If the program is supplied by DG, report in detail the exact steps used to reproduce the problem. If the program is supplied by another vendor or written by an installation, include a copy of the program and its source code if possible. Again, report in detail the exact steps used to reproduce the problem.

If your system panics, hangs or halts, see the DG/UX 5.4 release notice for instructions on taking system dumps and submitting DG/UX STRs.

<div align="center">End of DG/UX Window System Update Notice</div>