# GL2-T2.4 and GL2-T3.4 Terminal Release Notes

*Version 1.0*

**Silicon Graphics, Inc. 2011**
**Stierlin Road**
**Mountain View, CA 94043**

**Document Number 007-3501-010**

### Conventions

This document uses the standard UNIX convention for referring to entries in the UNIX documentation. The entry name is followed by a section number in parentheses. For example, *cc*(1) refers to the *cc* manual entry in Section 1 of the *UNIX Programmer's Manual.*

In command syntax descriptions and examples, square brackets ([ ]) surrounding an argument indicate that the argument is optional. Words in *italics* represent variable parameters, which should be replaced with the string or value appropriate for the application.

In text descriptions, filenames and UNIX commands are written in *italics*. IRIS Graphics Library and PROM monitor commands are written in `typewriter font`.

**Acknowledgments:**

Marcia Allen
Gail Kesner
Susan Luttner
Diane Wilford

Special thanks to Susan Ellis, Darrah Brown, Rick Nahass and Donl Mathis.

The information in this document is subject to change without notice.

**GL-T2.4 and GL-T3.4 Terminal Release Notes**
**Document Number: 007-3501-010**

# GL2-T2.4 and GL2-T3.4 Terminal Release Notes

These release notes describe the features of software release GL2-T2.4 for the IRIS 2000, 2200, and 2300 terminals and GL2-T3.4 for the IRIS 2300 Turbo and 3010 terminals. The seven sections of this document cover these topics:

1. Major Enhancements
2. Extent File System on the IRIS 2300
3. Installing Software Updates and Options
4. Additions to Graphics, Systems, and Options
5. Changes to Graphics, Systems, and Options
6. Bug Fixes to Graphics, Systems, and Options
7. Known Problems

Because of the increased functionality provided in this release, you must recompile all programs after installing this release.

## 1. Major Enhancements

The GL2-T2.4 and GL2-T3.4 terminal releases provide these major enhancements:

- The Extent File System, an improved file system with four to six times the performance of the standard UNIX System V file system (see Section 2, below). This is an enhancement for IRIS 2300 terminals only. The IRIS 2300 Turbo and 3010 already use the Extent File System.
- Many improvements to the Window Manager, *mex*(1G), including support for pop-up menus and multiple windows per process. This applies to the IRIS 2300 Turbo and 3010 only.
- Enhanced support for the IBM link, including the ability to transfer binary and text files between an IRIS and an IBM VM/CMS host. This enhanced support for the IBM link was released in GL2-T3.3.1 for the 2300 Turbo and 3010. This is applicable only to the IRIS 2300.

## 2. Extent File System

The IRIS 2300 terminal supports two file systems: the standard UNIX System V file system (also called the Bell file system) and the Extent File System (EFS).

EFS is installed when you install the GL2-T2.4 terminal release software on the IRIS 2300 (see Section 3).

EFS offers a significant improvement in file handling performance over the System V file system. Under the System V file system, data blocks are stored almost randomly on the disk. In reading or writing a file, the System V file system typically performs a seek for each block of recorded data. A considerable amount of overhead results each time a data block is accessed. In contrast, EFS uses a larger block size and stores groups of blocks (extents) contiguously on the disk. When files are stored contiguously, only one seek is required to access a file, reducing the access overhead. The larger block size results in the transfer of more data in a single disk access.

While offering these increases in performance, EFS is completely compatible with the System V file system from the user's point of view. The user interface is identical, and existing programs will work without change. Read/write operations and the use of *stdio* are identical. If you use *stdio*(3S), you need only recompile your program under the new release to improve program performance. If you use *read/write*, you can take full advantage of the new file system by increasing your buffer size to at least 4K.

## 3. Installing Software Updates and Options

This section contains the following procedures:

- Installing GL2-T2.4 software updates on the IRIS 2300
- Installing GL2-T3.4 software updates on the IRIS 2300 Turbo and 3010
- Installing host software updates

| NOTE |
|---|
| If you have a problem during the update procedure, resolve it before continuing. Failure to perform certain steps successfully can render the system unusable. If you encounter a problem that you cannot resolve, contact the Geometry Hotline at 800/345-0222 within California (toll-free), 800/252-0222 elsewhere in the U.S. (toll-free), or 800/443-0222 in Canada (toll free). |

### 3.1 Installing Software Updates on the IRIS 2300

The following procedure explains how to install the GL2-T2.4 software update on your IRIS 2300 terminal and supercedes the procedures in the *IRIS Terminal Guide, Series 2000*. This software is supplied on a set of floppy diskettes. The installation procedure upgrades the file systems on the disk to Extent File System (EFS), and then loads a complete new set of software from floppies. The installation commands provide status messages throughout the installation procedure.

The new standard software is on these floppies:

- **Bootstrap System**
- **EFS Upgrade Tools**
- **Standard System** floppies 1 through 6
- **Standalone**

If you ordered optional communication software, it is on these floppies:

- **IBM**
- **TCP1** and **TCP2**
- **IEEE 488**

To install the new software, follow these steps:

1. Back up any user files that you want to keep. Everything on the disk will be lost when the new file system is created. Include any user files that you have placed on the disk that were not shipped as part of the original system. Also, back up any of the following configuration files, if you have modified them.

   */.cshrc*
   */.login*
   */.profile*
   */.wsirisrc*
   */etc/checklist*
   */etc/gettydefs*
   */etc/group*
   */etc/getdate*
   */etc/passwd*
   */etc/sys_id*
   */etc/TZ*
   */etc/termcap*
   */tcpwsirisrc*

   If the TCP/IP communications option is installed, save the */etc/hosts* and */etc/hosts.equiv* files under a different name, such as */etc/hosts.old* and */etc/hosts.eq.old*. Back up these files with the configuration files. This allows you to restore them later without overwriting the new versions. When you restore the old versions of these files, compare the old with the new, and propagate your site-dependent information to the new versions. Do not directly replace the new versions of these files with your old ones.

Do not include any files on the backup other than user files that you have introduced onto the disk and the configuration files listed above. Otherwise, when you restore the files, you will overwrite the new software with the old.

You can either back up your files onto floppies, or, depending on your configuration, transfer the files temporarily to another machine. Back up user files onto floppies with the *tar* (1) command. Specify the list of file names you want to back up. For example, to back up the */etc/passwd* and */etc/sys_id* files, issue this command:

```
tar cvfb /dev/mf0a 80 /etc/passwd /etc/sys_id
```

2. Once you are confident that all required files are properly backed up, become the super-user and reboot the system:

   ```
   su
   reboot -q
   ```

3. Put the **Bootstrap System** floppy in the drive, and boot from it by typing:

   ```
   b mf0a:
   ```

   Ignore the following messages:

   ```
   WARNING: no root drive found during configuration
   INIT: Cannot open /etc/inittab. errno 2
   ```

   When the system prompts:

   ```
   ENTER RUN LEVEL (0-6, s, or S):
   ```

   type **s** and press RETURN.

4. Copy the bootstrap system to the **c** partition of the hard disk, and reboot from it:

   ```
   /etc/flpuxfer md0c
   reboot
   ```

```
b md0c:
```

Ignore the following messages:

```
INIT: Cannot open /etc/inittab. errno 2
```

When the system prompts:

```
ENTER RUN LEVEL (0-6, s, or S) :
```

type `s` and press RETURN.

5. When the system comes up, it prompts you to put the **EFS Upgrade Tools** floppy in the drive.

   Insert the **EFS Upgrade Tools** floppy in the drive and press RETURN. A new, empty Extent File System is created on **md0a** and the following message appears:

   ```
   EFS on /dev/md0a ready
   ```

6. Remove the **EFS Upgrade Tools** floppy.

7. Make a directory on which to mount the floppy device:

   ```
   mkdir /inst
   ```

8. Mount the first of the set of **Standard System** floppies and issue the *Build* command:

   ```
   mount /dev/floppy /inst
   /inst/Build
   ```

   Watch the screen for error messages and/or special instructions from these commands.

9. After loading the new software from the first floppy, the system tells you which floppy to insert next. Remove the floppy when prompted and insert the next of the set. As the system prompts you, continue removing and inserting floppies until you have inserted all six of them. Each time you insert a floppy, mount it and issue the *Build* command:

   ```
   mount /dev/floppy /inst
   /inst/Build
   ```

   The *Build* command checks whether all prerequisite floppies have already been installed before each floppy is read. If any are missing, it displays a message with the part number of the missing floppy and aborts. If this happens, insert the missing floppy and issue the *mount* and *Build* commands as shown above. The standard system is loaded when you see this message:

```
Software options may now be loaded.
```

Ignore the message:

```
a.out file = /root/recover
cannot open core
ready
_rootfs: xFF = 0x2
_nswap:  00 = 0x2FD0
```

10. Reboot the system:

```
reboot
b
```

If you are using a 30 Hz monitor, enter:

```
reboot
b mon
b md0a:
```

11. The system will ask you which host to connect to. Become the superuser:

```
~!
su
```

12. If you have ordered any communication software, follow the procedure in steps 7 through 10 to load it. After the software is loaded, remove the installation directory:

```
rmdir /inst
```

13. Restore the user and configuration files that you backed up in step 1. Restore the files with this command:

```
tar xvfb /dev/mf0a 80
```

14. Copy the system onto the backup partition:

    ```
    /etc/rootcopy ok
    ```

See the procedure for making backup floppies in your *IRIS Terminal Guide, Series 2000 Update Package, Version 1.1*.

### 3.2 Installing Software Updates on the IRIS 2300 Turbo and 3010

The following procedure explains how to install the GL2-T3.4 terminal software on an IRIS 2300 Turbo or 3010 running a previous release. This new software is distributed on floppy diskettes.

Before installing updates, back up any files on the disk that you have changed. Make sure you have a bootable backup floppy set on hand. (See the section on Creating and Using a Backup Floppy Set in the *IRIS Series 3000 Owner's Guide*.)

The new standard software is on these floppies:

- **Standard System** floppies 1 through 6

If you ordered optional communication software, it is on these floppies:

- **IBM**
- **TCP1** and **TCP2**
- **IEEE 488**

To install the new software, follow these steps:

1. Start a shell and become the super-user:

    ```
    su
    ```

2. Make a directory on which to mount the floppy device:

    ```
    mkdir /inst
    ```

3. Verify that */defaultboot* and */vmunix* are linked to */kernels/3000.std* before installing the update. Issue these commands:

    ```
    rm /defaultboot /vmunix
    ln /kernels/3000.std /vmunix
    ln /vmunix /defaultboot
    ```

4. Put the first **Standard System** floppy in the floppy drive. Mount the floppy and issue the *Update* command:

```
mount /dev/floppy /inst
/inst/Update
```

The *Update* command checks whether all prerequisite floppies have already been installed before each floppy is read. If any are missing, it displays a message with the part number and aborts. If this happens, insert the missing floppy and issue the *mount* and *Update* commands as shown above.

5. The *Update* command may display other error messages or instructions. Watch the screen for such messages.

   While reading the floppy, the system displays a list of configuration file names. For each of these, the old version is stored with an equal sign (=) appended to its name, so after the installation, you can compare the new version with the old. This helps you to transfer any site-specific information from the old version to the new version after the installation is complete.

   After loading the new software from the floppy, the system tells you which floppy to insert next. Remove the floppy when prompted and insert the next of the set. Each time you mount a floppy, issue the *Update* command:

   ```
   mount /dev/floppy /inst
   /inst/Update
   ```

   As the system prompts you, continue removing and inserting floppies until you have inserted all six of them.

6. Once all software is loaded, reboot the system:

   ```
   reboot
   b
   ```

7. If you have ordered any communications option, follow steps 4 through 5 to load it. Verify that */defaultboot* and */vmunix* are linked to */kernels/3000.std* before loading the communications floppies. Issue these commands:

   ```
   rm /defaultboot /vmunix
   ln /kernels/3000.std /vmunix
   ln /vmunix /defaultboot
   ```

   Substitute the communications floppies for the **Standard System** floppies.

8. Make site-specific changes to the configuration files reported by the *Update* command. Restore any user files you backed up by issuing this command:

   ```
   tar xvfb /dev/md0a 80
   ```

### 3.3 Host Software Update Installation

Section 3.3.1 contains instructions for installing host-resident terminal software in a Unix 4.2BSD environment. Section 3.3.2 contains instructions for installing host-resident terminal software updates on an IRIS host.

#### 3.3.1 Installing GL2-T2.4 and GL2-T3.4 Software on a UNIX 4.2BSD System

The Unix 4.2BSD host software for the IRIS terminal is distributed on a 1600 bpi magnetic tape in *tar*(1) format. This tape contains two types of software:

- host software updates that reside on the host computer
- terminal software for the IRIS 2000 and 2200 that must be downloaded from the host computer onto the IRIS terminal each time the IRIS terminal is booted

The new software replaces the existing software. Customers who have a floppy drive on their IRIS 2000 or 2200 receive a floppy disk containing the terminal software.

The host software can be stored anywhere on the host file system. For purposes of explanation, the examples below assume that the host software will be installed in a directory called */usr/iris*.

1. Log in to UNIX as root.
2. Check whether you have an existing directory named *usr/iris*. If so, remove its contents:

   ```
   rm -rf /usr/iris
   ```

   If not, create a directory to hold the host software:

   ```
   mkdir /usr/iris
   ```

3. Change directories to the new directory */usr/iris*:

   ```
   cd /usr/iris
   ```

4. Load the distribution tape onto a tape drive and read the software into the new directory:

   ```
   tar xv
   ```

5. Change directories to */usr/iris/c*:

   ```
   cd /usr/iris/c
   ```

   Edit the file */etc/termcap*. Add the file *iris.termcap*, which contains the terminal description for the IRIS terminal.

   If you use FORTRAN, change directories to */usr/iris/f77*:

   ```
   cd /usr/iris/f77
   ```

6. If you are loading software to be used on an IRIS 2000 or 2200, define the shell variable *IRISBOOT* to be the directory that contains the IRIS terminal program *iris*:

   *for sh users:*

   ```
   IRISBOOT=/usr/iris/boot
   export IRISBOOT
   ```

   *for csh users:*

   ```
   setenv IRISBOOT /usr/iris/boot
   ```

7. Compile the Remote Graphics Library (*libgl.a*), the serial download program (*dliris* for the IRIS 2000 and 2200), and the demonstration programs:

   ```
   make all
   ```

8. To use the automatic network boot environment for booting the terminal, link the file that is to be the default boot file to *defaultboot* or *defaultboot0*. For example, when *iris* is the default boot file, type:

   ```
   cd /usr/iris/boot
   ln iris defaultboot
   ```

   *or*

   ```
   cd /usr/iris/boot
   ln iris defau1tboot0
   ```

   > **NOTE:** Make sure the complete path name is specified when using *sgboot*(1) and *sgbounce*(1).

### 3.3.2 Installing GL2-T2.4 and GL2-T3.4 Host Software on an IRIS

This section contains instructions for installing host-resident software on an IRIS workstation. The new software replaces the existing software. The software for IRIS workstation hosts is distributed on 1/4" cartridge tapes in Silicon Graphics' standard distribution format. Included on the tape are a set of commands that help install the software. These commands and some associated files are referred to as "Distribution Tools".

#### Preparation

Before installing new host software on the IRIS, first edit the file *rc.xns*. Remove the pound signs (#) at the beginning of the two lines that contain *sgbounce*. Change the line that reads */usr/local/boot* to read */usr/boot*. Then reboot the system.

#### Installing from Tape

To install the host software, follow the steps below. The basic procedure assumes that the host has a 1/4" tape drive. If the tape drive is on a remote machine to be accessed using XNS, see "Network Installation" at the end of this section.

1. Log in as root. Make sure the system is in multi-user mode, but that there are no other users logged on.

2. Change your working directory to the *root* directory:

```
cd /
```

3. If an old copy of the distribution directory is present, remove it:

```
rm -rf dist
```

4. Put the distribution tape in the tape drive.
5. Read in the distribution tools with *cpio*:

```
cpio -ivhmud1
```

6. Read in the files from the tape:

```
/dist/Read
```

   If you would like feedback while the command progresses, give the **–v** (verbose) option:

```
/dist/Read -v
```

   The **v** option of the *Read* command shows the names of the files as they are extracted from the tape. The disadvantage of the verbose option is that error messages can scroll off the screen unnoticed.

7. Perform the installation details:

```
/dist/Install
```

   This command reports its progress and the names of the affected files.

8. Delete the distribution directory:

```
rm -rf /dist
```

### Network Installation

If you are installing the distribution on a workstation that does not have a tape drive, use a remote workstation with a tape drive and the XNS network protocol.

Follow the procedure above, but with these changes:

1. Log in as root on the remote machine. Follow steps 2 through 5 as above.

   Log in as root on the machine on which you are installing the software. Copy the distribution tools to the machine on which you are installing the software using these commands (*remote* is the name of the machine with the tape drive):

   ```
   cd /
   xcp -r -v remote:/dist .
   ```

2. Issue the *Read* command with the **–x** option to tell the *Read* command the name of the remote machine.

   ```
   /dist/Read -x remote
   ```

3. Continue with steps 6, 7, and 8 above.

## 4. Additions

The next three sections describe additions to graphics, systems, and options software for terminals. Unless otherwise indicated, the additions listed here apply to all IRIS terminals.

### 4.1 Graphics Additions

This section lists additions to the Remote Graphics Library since GL2-T2.3.1 for the IRIS 2000, 2200, and 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** — **Description**

`blanktime()` — A new function `blanktime(nframes)` sets the duration before the screen is blanked. If *nframes* is 0, the screen never blanks. This addition applies to the IRIS 2300, 2300 Turbo, and 3010.

*device.h* — A set of timer devices, TIMER0, TIMER1, TIMER2, and TIMER3, were added to the file *device.h*. These can be queued and read like valuators. Setting noise on a queued timer specifies how often the timer should add events to the input queue. This addition applies to the IRIS 2300, 2300 Turbo, and 3010.

A MENUBUTTON device, added to *device.h*, is defined to be the same as the right mouse button. Applications that use pop-up menus should use MENUBUTTON. This addition applies to the IRIS 2300, 2300 Turbo, and 3010.

`getdev()` — A new command `getdev()` allows more than one valuator to be sampled in a single call. This addition applies to the IRIS 2000, 2200, and 2300.

`getothermonitor()` — The new function `getothermonitor()` returns the display mode available and not currently in use.

`ismex()` — The `ismex()` routine returns TRUE if the window manager, *mex*, is running. This addition applies to the IRIS 2300, 2300 Turbo, and 3010.

`isqueued()` — A new function `isqueued(device)` returns TRUE if the specified device is queued.

*pop-up menus*

A simple pop-up menu package allows user processes to use the same pop-up bitplanes that the window manager uses. The Remote Graphics Library now includes the following routines:

`addtopup()` adds items to an existing pop-up menu.

`defpup()` defines a pop-up menu.

`dopup()` calls up the indicated pop-up menu.

`endfullscrn()` ends full screen mode, returning the screen mask and its viewport to the bounds of the current graphics port.

`endpupmode()` ends pop-up mode.

`fullscrn()` enables the entire screen for writing under the window manager.

`freepup()` returns a menu and structures to the system.

`newpup()` allocates and initializes a structure for a new menu.

`pupcolor()` specifies the current pop-up drawing color.

`pupmode()` provides access to the pop-up menu bit planes.

`winattach()` attaches the input focus to the current graphics port and calling process.

`winclose()` closes the identified graphics port.

`winconstraints()` changes the constraints associated with the current graphics port.

`winmove()` changes the screen location of the current graphics port.

`winopen()` creates a graphics port under the window manager.

`winpop()` moves the current graphics port in front of all other ports.

`winpush()` moves the current graphics port in back of all other ports.

`winset()` makes the identified port current.

`wintitle()` adds a title bar to the current graphics port.

These routines were added to the IRIS 2300, 2300 Turbo, and 3010.

`rcrv(), rcrvn(), rpatch()` Rational cubic routines `rcrv()`, `rcrvn()`, and `rpatch()` were added. These are rational versions of the `crv()`, `crvn()`, and `patch()` commands.

`rot()` The new `rot()` command rotates floating point angle arguments.

### 4.2 Systems Additions

This section lists additions to Unix since GL2-T2.3.1 for the IRIS 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** **Description**

*/etc/inittab* A *sysint* state has been added to */etc/inittab*. It executes a new script */etc/rc.s0* that resets the virtual system console to the actual physical system console. This addition applies to the IRIS 2300, 2300 Turbo, and 3010.

### 4.3 Additions to Options

This section lists additions to optional peripherals and communications options since GL2-T2.3.1 for the IRIS 2000, 2200, and 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** **Description**

*dial and button box* Dial and button boxes and digitizer tablets can now be connected to ports 2, 3, or 4. A new Graphics Library routine, `devport()`, was added to support this connection.

*IBM Link*

The IBM link supports binary and text file transfer between an IBM VM/CMS host and an IRIS. This addition applies to the IRIS 2300.

*IRIS Terminal Software Option*

A new IRIS Terminal Software Option supports an IRIS series 3000 (or Turbo) workstation host and a series 2000 terminal.

A new IRIS Terminal Software Option supports an IRIS series 2000 workstation host and a series 3000 terminal.

light pen

A light pen option was added. A device, LPENVALID, acts like a button to indicate when the light pen position is valid. This addition applies to the IRIS 2300, 2300 Turbo, and 3010.

*wsiris*

If the **–y** command line option is given to *wsiris*, it will abort if put in fast mode.

## 5. Changes

The next three sections describe changes to graphics, systems, and options software for terminals. Unless otherwise indicated, the changes listed here apply to all IRIS terminals.

### 5.1 Graphics Changes

This section lists changes to the Graphics Library since GL2-T2.3.1 for the IRIS 2000, 2200, and 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** — **Description**

`chunksize`()

In earlier releases, the error message

```
ERR SIZEFIXED
```

was reported if `chunksize`() was called after the chunksize was frozen. Now

```
CHUNKSIZE IS FROZEN
```

is reported.

`depthcue`()

Polygons and characters, as well as lines, are now depth-cued.

*European video standard*

*get.h*, `setmonitor`(), and `getmonitor`() now handle the PAL display option.

*graphical objects*

The method of freeing graphical objects has been improved. You are now less likely to run out of memory during long graphics sessions.

*mex*(1G)

All *mex* changes apply to the IRIS 2300, 2300 Turbo, and 3010.

The cursor shape now belongs to the process that currently has the input focus.

New command line options **–d** and **–t** were added. **–d** causes the window manager to remain in double buffer mode. **–t** allows the window manager to accept a font for titles.

When a window is created or moved, the new region is no longer cleared to black. Windows can now slide off the screen. Window size and shape can be controlled from a process.

*mex* now supports multiple windows per process.

When *mex* puts itself in the background, it now exits *with* a return code of 0 rather than –1.

`imakebackground`() is now a window manager hint, just like `winaspect`() or `prefposition`(). `imakebackground`() must be followed by `winopen`().

Events are distributed to applications and the window manager in a new way. The current convention makes the right mouse button the menu button. All input events, other than the menu buttons, from buttons and valuators are delivered to the window that currently has the input focus. Input from menu buttons is handled according to these rules:

- If the cursor is over the window that currently has the input focus and this window has the right button queued, a button-down event is handed to this window. Applications that provide pop-up menus use this feature. Otherwise, the button-down event goes to the window manager.
- If the cursor is over the background and the menu button is pressed, a short menu appears.
- If the cursor is over another window, a longer menu appears allowing that window to be moved, pushed, popped, etc.
- If the cursor is pressed in the header of the window at any time, it brings up a menu that allows that window to be moved, pushed, popped, etc.

To access the pop-up menus provided by applications, you must not have a RESERVEBUT entry for the right mouse button in your *.mexrc*.

**NOTE:** If titles do not appear correctly on your windows in *mex*, examine the */.mexrc* file.

Look at the “bindindex” lines. The color index for the background of the title bar is specified by *inborder* and *hiinborder*, while *titletextout* and *hititletextout* specify the color index for the horizontal lines drawn in the title bar. *titletextin* and *hititletextin* are used for the text in the title. The names starting with “hi” are used when the input focus is attached to a window.

*objects and curves*

Object code and curve code are now loaded only if objects and curves are used. This makes many graphics programs smaller. This change applies to programs running locally on the IRIS 2300, 2300 Turbo, and 3010.

*screen unblanking*

The screen now unblanks whenever any valuator moves or any button changes state. Before, only the keyboard and the mouse caused the screen to unblank. This change applies to the IRIS 2300, 2300 Turbo, and 3010.

*shaded polygons*

Shaded polygons paint faster than in the GL2-T2.3.1 release.

### 5.2 Systems Changes

This section lists changes to Unix since GL2-T2.3.1 for the IRIS 2000, 2200, and 2300, and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** — **Description**

*alternate keypad mode* — You now enter alternate mode with **ESC ;** or **ESC =** . The key codes transmitted are now VT52 standard.

| Key | Alternate keypad code |
|---|---|
| 0 | ESC ? p |
| 1 | ESC ? q |
| 2 | ESC ? r |
| 3 | ESC ? s |
| 4 | ESC ? t |
| 5 | ESC ? u |
| 6 | ESC ? v |
| 7 | ESC ? w |
| 8 | ESC ? x |
| 9 | ESC ? y |
| - (dash) | ESC ? m |
| , (comma) | ESC ? l |
| . (period) | ESC ? n |
| ENTER | ESC ? M |
| PF1 | ESC P |
| PF2 | ESC Q |
| PF3 | ESC R |
| PF4 | ESC S |

You can escape from alternate keypad mode by pressing the **ESCAPE** and > keys simultaneously.

*wsiris* — All *wsiris* changes apply to the IRIS 2300, 2300 Turbo, and 3010.

*wsiris* error messages now give the name of the Graphics Library command being processed when appropriate.

New keyboard commands **~%Dn** and **~%Zn** toggle the **–d n** and **–z n** options. (See *wsiris* (1C)). **~%D3** takes the characters following it up to a terminating **RETURN** to be the new logfile name. The current one is then closed and the new one opened for appending. If the `RETURN` follows immediately, the current logfile name is displayed.

*wsiris* has a new keyboard command: press ~ and `DEL`. This reboots the IRIS after asking for confirmation.

There is a new **–d** command line option. **–d 3** *filename* sets the logfile name to *filename*. With **–d 5**, the *transmit* process is killed when initializing the graphics. This is useful when debugging the *receive* process.

When running *mex* in *wsiris*, `winattach`() is called by `ginit`(), `gbegin`(), `getport`(), or `winopen`(). This is done so that the graphics window, which is always in the foreground, is attached automatically.

If you type a hostname incorrectly, you can now interrupt out of a connection attempt in *wsiris* without waiting for the connection failure.

Floating-point errors are now caught using *fpsigset*. A message is displayed and processing continues. Previously, *wsiris* dumped core.

FORTRAN routines with character array arguments now appear in the logfile as strings instead of arrays. The argument to `dbtext`() is now logged as a string.

*wsiris* checks the return status of the call to *system*, which does the TCP/IP initialization. If the initialization failed, it prints a message and exits.

### 5.3 Changes to Options

This section lists changes to the FORTRAN tools and to communications options since GL2-T2.3.1 for the IRIS 2000, 2200, and 2300, and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

| **Item** | **Description** |
|---|---|
| */etc/inittab* | The **ttyp[0-3]** devices in */etc/inittab* have been changed to **ttyT[0-7]**. This change applies to the IRIS 2300, 2300 Turbo, and 3010. |
| *FORTRAN* | *fgl.h* now contains a NAMEST parameter that tells the depth of the name stack. |
| *IBM* | Users can now escape *t3279* to a Bourne shell. This change applies to the IRIS 2300.<br>*mT3279* was removed, since its functions have been incorporated into *t3279*. This change applies to the IRIS 2300 Turbo and 3010. |
| *IEEE 488* | */etc/ibtab* now initializes an additional IEEE 488 device. This change applies to the IRIS 2300. |
| *irsh* | The real user-id of the command executed on the terminal is now root. |
| *TCP/IP* | All TCP/IP changes apply to the IRIS 2300, 2300 Turbo, and 3010.<br>TCP/IP special files */dev/ttyT?* replace */dev/pty?* and */dev/typ?*. */etc/inittab* and */etc/ttytype* have been updated to reflect this name change.<br>The TCP/IP *netload* utility provides new protocol module configuration options. In particular, the local host is selected by the alias “localhost” in */etc/hosts*. |

## 6. Bug Fixes

The next three sections list the bug fixes in IRIS graphics, system, and optional software for terminals. Unless otherwise indicated, the additions listed here apply to all IRIS terminals.

### 6.1 Graphics Bug Fixes

This section lists bug fixes in the Remote Graphics Library since the GL2-T2.3.1 release for the IRIS 2000, 2200, and 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** **Description**

`backface()` The `backface()` command now correctly handles rectangles made with `rectf()`.

`crv()` `crv()` no longer causes extraneous lines to be drawn.

`crvn()` `crvn()` in a picked object no longer generates the error message:

```
No feedback in pickmode in getmatrix
```

*depth cue* Entering depth cuing while in pick mode works correctly.

*feedback* `readpixels()`, `getmatrix()`, and `loadmatrix()` now work in feedback mode.

*font RAM* The font RAM is no longer overwritten unintentionallv.

`getdev()` The `getdev()` command now works remotely. This bug fix applies to the IRIS 2300, 2300 Turbo, and 3010.

`gexit()` Keypad mode is now restored with `gexit()`, if the keypad mode is turned on and a graphics program is run on a terminal other than the console.

`gexit()` is no longer ignored if `gbegin()`, `getport()`, or `winopen()` are used instead of `ginit()`. This bug fix applies to the IRIS 2300, 2300 Turbo, and 3010.

*keyboard buttons* Both the keyboard buttons and the keyboard are unqueued on exit, so that even on an exception or interruption, the keyboard is usable. This bug fix applies to the IRIS 2300, 2300 Turbo, and 3010.

*mex*(1G)

All *mex* bug fixes apply to the IRIS 2300, 2300 Turbo, and 3010.

A call to `setcursor()` to change the cursor glyph no longer causes the cursor to disappear and not reappear until mex sets it. `setcursor()` now ignores its color and write mask arguments when *mex* is running.

Running the *gclear* command after exiting *mex* now works correctly.

Pop-up menus near the edge of the screen now work correctly.

*mex* now handles mouse buttons correctly when the NOSCROLL key is used in “hogwhiledown” mode.

All keyboard input is sent to the attached process.

*mex* now maintains the requested aspect ratio when the mouse runs into the edge of the screen.

Multiple windows in a double-buffered *mex* program no longer confuse the display mode.

*mex* no longer dumps core if `keepaspect()` is called with no arguments.

`objreplace()`

Incorrectly freed memory no longer causes an `objreplace()` bug related to the `crv()` and `patch()` commands.

`popattributes()`

The current font is now restored by `popattributes()`.

`rectcopy()`

Several `rectcopy()` bugs are fixed. These bugs had appeared when you called this command in the window manager or used it with some copy directions. The screen coordinates are now relative to the physical screen instead of to the lower left corner of the graphics port. These bug fixes apply to the IRIS 2300, 2300 Turbo, and 3010.

`shaderange`()

The C and FORTRAN parameter type declarations are now correct. This bug fix applies to the IRIS 2000, 2200, and 2300.

`textport`()

The textport now handles tabs that wrap past the end of a line.

The call `textport`(0,0,0,0) no longer causes a kernel bus error panic after a few keystrokes.

*wsiris*

All *wsiris* bug fixes apply to the IRIS 2300, 2300 Turbo, and 3010.

%**M** now echoes the correct monitor type.

*wsiris* now recognizes `gbegin`(), get `port`() and `winopen`() as graphics initialization routines.

**6.2 System Bug Fixes**

This section lists bug fixes to Unix and XNS since the the GL2-T2.3.1 release for the IRIS 2000, 2200, and 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

**Item** **Description**

*block mode I/O*

Block mode serial I/O is included in the kernel. This bug fix applies to the IRIS 2300 Turbo and 3010.

*disk model number*

The model number of the Tandon 20 Mb disk, buried in kernel files *mdfex*, *sgilabel*, and *uname*, is now correct. It has been changed to TM-262. This bug fix applies to the IRIS 2300.

*/etc/termcap*

The */etc/termcap* entries for the underline start and underline end sequences for IRIS systems are now correct. This bug fix applies to the IRIS 2300, 2300 Turbo, and 3010.

*phys*(2)

You must be the super-user to use this system call. *phys* allows a process to access physical addresses. This bug fix applies to the IRIS 2300.

*xlogin*(1)

XON/XOFF no longer causes inappropriate delays when you log in remotely to a workstation. This bug fix applies to the IRIS 2300, 2300 Turbo, and 3010.

### 6.3 Bug Fixes to Options

This section lists bug fixes to peripheral options and communications options the GL2-T2.3.1 release for the IRIS 2000, 2200, and 2300 and since the GL2-T3.3.1 release for the IRIS 2300 Turbo and 3010.

| **Item** | **Description** |
|---|---|
| `depthcue`() | Depth cuing in FORTRAN now works. |
| *dials and buttons box* | The first `dbtext`() string sent to the LEDs on the dials and buttons box after bootup is now printed. This bug fix applies to the IRIS 2300. |
| *digitizer tablet* | The file *device.h* now defines the digitizer tablet devices. This bug fix applies to the IRIS 2000, 2200, and 2300. |
| */etc/rc.tcp* | The */etc/rc.tcp* bug fixes apply to the IRIS 2300.<br>*/etc/rc.tcp* no longer requires that the local host be the first entry in */etc/hosts*.<br>*/etc/rc.tcp* now reports when the *netload* failed. |
| *FORTRAN* | The FORTRAN version of `shaderange`() now has the correct number of parameters. This bug fix applies to the IRIS 2000, 2200, and 2300.<br>The FORTRAN Remote Graphics Library version of the `winopen`() command now returns a value. This bug fix applies to the IRIS 2300, 2300 Turbo, and 3010. |
| *IBM Link* | */dev/pxd*, required by the IBM link, is included in the link software. This bug fix applies to the IRIS 2300. |
| *IEEE 488* | All IEEE 488 bug fixes apply to the IRIS 2300, 2300 Turbo, and 3010.<br>The IEEE 488 driver now handles odd-length writes correctly.<br>When using the IEEE 488 interface, sequences of short (one- or two-byte) transfers no longer fail randomly.<br>When IEEE 488 is used for graphics and serial line text, all of the child processes are killed after *wsiris* exits. |

The IEEE 488 device with *iib* **–r** no longer requires resetting.

*iftp*

`iftp  w` no longer fails when the destination directory is not writeable by *guest*, e.g., /.

*TCP/IP*

All TCP/IP bug fixes apply to the IRIS 2300, 2300 Turbo, and 3010.

TCP now follows TIME-WAIT specifications. In the past, TCP did not close TCP connections promptly, which violated specifications across networks with a long round-trip time.

The TCP/IP *telnet* server has improved its processing of AYT. It returns a `CTRL-G` directly to the user and does not emulate a `CTRL-G` in the user's input stream.

Mismatches between advertised and actual header/data lengths in incoming packets are now properly accommodated.

TCP/IP *select*() works on UDP sockets.

The TCP/IP *telnet* utility sends a `< cr > < lf >` sequence, as recommended by the TELNET specification. Previously, it sent only a `< cr >`.

*wsiris*

All bug fixes in the *wsiris* section apply to the IRIS 2300, 2300 Turbo, and 3010.

*wsiris* no longer hangs if a bad host IP address is entered. IP address syntax checking is now done with three periods (...).

Using IEEE 488 in graphics and serial lines for the textport now works.

*wsiris* now works with IEEE 488.

When using IEEE 488 for graphics and serial line for text, all of the child processes are killed when *wsiris* is exited.

## 7. Known Problems

This list contains known problems with this release, and, where possible, suggested workarounds.

*autoboot*

After installing GL2-T2.4, you cannot use the autoboot function with a 30 Hz monitor.

`depthcue`()

A small number of pixels just outside a shaded range of colors may appear in an image where lines are clipped by the far clipping plane. A workaround is to extend the color range to include the overflow bits, or to reduce the values passed to shaderange. For example, if indices 128-191 are used for a shaded range of colors, and pixels appear outside that range, call

```
shaderange(135,185,...)
```

or extend the color range to 125-196 and call

```
shaderange(128,191,...)
```

*font RAM*

When large fonts are repeatedly defined and deleted, the space in the font RAM is sometimes not cleaned up correctly. This may preclude another font from being defined correctly, even though there should be enough space to hold it.

*floppy incompatibility*

A floppy formatted on a 3030 is not readable on a 3010. A floppy formatted on a 3010, however, is readable and writable on a 3030. To transfer information via floppy from a 3030 to a 3010, format the floppy on the 3010, then write to it on the 3030. Information can also be transferred from a 3030 to a 3010 over a network, and then saved to a floppy on the 3010.

`getmem`()

`getmem`() sometimes returns the wrong value.

*lines*

Wide lines do not work correctly in depth cue mode.

*mex*(1G)

The `textcolor`() and `pagecolor`() commands do not work correctly under *mex*. Sometimes the color of the wrong textport is changed or the commands may appear to have no effect.

*phys*(2)

This system call is not supported in this release.

`pick`()

Picking sometimes fails when objects are nested to a depth of more than six or seven.

*TCP/IP*

An incoming *rsh* may hang if at the same time another *rsh* in either direction is starting or ending.

`zbuffer`()

A small number of pixels just outside a shaded range of colors may appear in an image. Use the same workaround described for `depthcue`(), above.