GL2-W2.4 Workstation Release Notes

Version 1.0

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

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

Special thanks to Susan Ellis and Donl Mathis.

GL2-W2.4 Workstation Release Notes
Document number: 007-3101-010

GL2-W2.4 Workstation Release Notes

These Release Notes describe the features of Software Release GL2-W2.4 for IRIS 2400 and 2500 workstations. The eight sections of this document cover these topics:

> 1. Major Enhancements
> 2. Extent File System
> 3. Shared Memory
> 4. Installing Software Updates and Options
> 5. Additions to Graphics, Systems, and Options
> 6. Changes to Graphics, Systems, and Options
> 7. Bug Fixes to Graphics, Systems, and Options
> 8. Known Problems

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

1. Major Enhancements

The GL2-W2.4 workstation release provides 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).

- System V shared memory (see Section 2 below).

- Improved virtual memory subsystem performance.

- Many improvements to the window manager *mex*(1G), including support for pop-up menus and multiple windows per process. Examples of window manager programs are contained in */usr/people/gifts/mextools*; see the *README* in that directory to use these programs.

- Enhanced support for the IBM link, including the capability to transfer binary and text files between an IRIS and an IBM VM/CMS host.

Version 1.0

- A local Pascal Graphics Library.
- Language enhancements:
  - You can make interlanguage calls between C and FORTRAN directly (without wrappers) in both directions. The C compiler storage class *fortran* now supports these calls. See the *C/FORTRAN Interface* document in the version 1.2 update of volume IIB of the *UNIX Programmer's Manual* which is distributed with this release .
  - The source-level debugger *dbx* now debugs FORTRAN programs. We provide a *dbx* tutorial (C and FORTRAN versions) for new users.
  - A new **+s** option to the FORTRAN compiler adds a SAVE statement to the beginning of each routine. This option causes variables local to a subroutine to retain their values between calls.
  - Pascal users now have the same control as FORTRAN users over floating-point exceptions. See *IRIS Floating-point* in the version 1.2 update of volume IIB of the *UNIX Programmer's Manual*.
  - The FORTRAN and Pascal compilers now produce standard UNIX binaries (*.o* files) rather than special binaries (*.j* files). You must recompile all FORTRAN and Pascal programs for this release.

    The default rules of *make*(1) reflect this change in object file names. You will need to update your makefiles. See the *C/FORTRAN Interface* (FORTRAN users) or version 2.0 of the *Pascal Release Notes for the Graphics Library* (Pascal users).

    The standard UNIX archiver *ar*(1) replaces the *library*(1) utility, eliminating library cross-referencing problems.
  - The new assembler *as*(1) is a version of the 68020 assembler. The input language for *as*(1) is limited to the 68010 instruction set and addressing modes. The old assembler is available for this release only in */bin/oas*. See the *Assembler Reference Guide* in the *UNIX Programmer's Manual, Volume IIB* for documentation of the new assembler. See the Systems Changes Section of this document for differences in the input language accepted by the old and new assemblers.

2.   Extent File System

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

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 to the System V file system, 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. 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 with the new kernel 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.

EFS is installed when you install the GL2-W2.4 workstation release software (see Section 4).

3. Shared Memory

The System V shared memory facility is now fully supported. Shared memory operates under these restrictions:

- The maximum shared memory size (SHMMAX) is 64kb, or 0x10000 bytes.

- The minimum shared memory size (SHMMIN) is 1 byte.

- The maximum number of shared memory segments per system (SHMMNI) is 40.

- The maximum number of shared memory segments per process (SHMSEG) is 10.

- The maximum number of shared memory pages per system (SHMALL) is 0x40, or 256kb of memory. The default separation between the end of *bss* space (see the manual page for *sbrk*(2)) and the first system-chosen shared memory segment address (SHMBRK) is 32768 bytes. This parameter controls where the system places a shared memory segment when you use the *shmat* command on the segment with an address of zero.

All these limits apply simultaneously. While a process may not use up its number of segments-per-process limit, it may still overflow the SHMMNI limit, or, more likely, the SHMALL parameter.

Under this implementation, the shared memory segments are not paged. Therefore, using many shared memory segments reduces the amount of memory for user programs. This may increase paging and slow down the system. By default, shared memory segments are not destroyed when a process exits. *ipcrm*(1) deletes shared memory segments.

Shared memory manipulation is described in these manual pages.

- *ipcrm*(1)
- *ipcs*(1)
- *shmctl*(2)
- *shmget*(2)
- *shmop*(2)

4.   Installing Software Upgrades and Options

The GL2-W2.4 distribution tape contains new standard system software, any software options you may have ordered, and tools used to update your file systems to EFS and install the new software. This section explains how to use the distribution tape to update your file systems and install the software release on your IRIS.

> **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 for Canada.

4.1 Preparation

Before installing the software update, it is essential to back up your user files. The entire disk will be rebuilt with a standard system, and all current contents of the disk will be lost. Create a backup tape that contains files that are unique to your system—your user files. Do not include the standard system files on the tape; otherwise, when you restore these files from tape, you will overwrite the new software you just installed. However, some configuration files must be backed up onto tape and restored later. Back up these configuration files:

> */etc/checklist*
> */etc/cshrc*
> */etc/gettydefs*
> */etc/group*
> */etc/passwd*
> */etc/profile*
> */etc/sys_id*

If you have TCP/IP communication option, copy these configuration files by issuing these commands:

```
cd /etc
cp hosts hosts.old
cp hosts.equiv hosts.eq.old
```

and back up */etc/hosts.old* and */etc/hosts.eq.old* onto tape.

If you have arranged your user files so that they are easily accessible (for example, in subdirectories under */usr/people*), simply back up these subdirectories. If you have installed user files in standard directories such as */usr/bin*, */usr/lib*, and so on, it may be more difficult to find them all. In either case, restrict the backup to the files that you have introduced onto the disk; exclude files that have been supplied either on the original system disk or on update or option tapes that have come from Silicon Graphics.

The following list contains the names of 24 standard files and directories supplied by Silicon Graphics. This list is not exhaustive and does not contain subdirectories and empty directories supplied by Silicon Graphics. Do not include these files and directories on your backup tape:

/Versions	*/bin*
/defaultboot	*/dev*
/etc	*/kernels*
/lib	*/stand*
/usr/adm	*/usr/bin*
/usr/diag	*/usr/dict*
/usr/games	*/usr/include*
/usr/lib	*/usr/man*
/usr/people/demos	*/usr/people/gifts*
/usr/people/guest	*/usr/people/mexdemos*
/usr/pub	*/usr/spool*
/vmunix	*/vmunix1*

You can use either *tar*(1) or *cpio*(1) to back up your files. Backup procedures are included in Chapter 4 of the IRIS Workstation Guide, Series 2000.

4.2 Distribution Tape Contents

The update distribution tape contains several images, each of which serves a specific function. An image is a collection of files that are written to the tape as a unit. The first image on the tape is a set of software distribution tools, residing in a directory called */dist*. These tools consist of commands and data files that are used to install software options.

Version 1.0

The next three images on the tape are named *boot*, *root*, and *usr*. The *boot* image is a special system that performs the EFS update procedure to make new file systems, and then loads the *root* and *usr* images onto the root (/) and */usr* file systems of the disk.

If you have ordered any optional software, each option is contained in an additional image on the tape. For example, if you have ordered the FORTRAN and TCP/IP options, they appear in images called *ftn* and *tcp* following the *usr* image.

The file */dist/toc* contains the table of contents of files on the tape. Once the distribution tools are read from tape onto the disk, examine this table of contents file to determine what images are included on the remainder of the tape. Each image has a shorthand name by which it is known to the distribution tools. These shorthand names are the names used in *toc*. To decode these names, examine the */dist/desc* file.

The update procedure accesses the *boot*, *root*, and *usr* images automatically. You are not required to name them explicitly. However, if you install any options, you need to give their names as arguments to some of the distribution commands. During the procedures that follow, you will be directed to examine the *toc* file at the appropriate time to determine the shorthand names of the options you want to read in from the tape.

4.3 Installing Software Updates in Workstations with a Tape Drive

The following instructions apply only to systems with a tape drive.

1. During this procedure, the entire contents of the disk will be destroyed and replaced with the new release of the standard system and software options. All files currently on the disk will be lost. Back up all user files that you want to keep, using *tar*(1) or *cpio*(1) to write them onto tape. Restrict the backup tape to user files and those configuration files listed above. Exclude all standard system files and directories supplied by SGI (see *Preparation*, above).

2. Perform a "table of contents" operation on the backup tape to ensure that it is readable. Both *tar* and *cpio* have a **t** option that does this. The *tar* command is:

    ```
    tar tv
    ```

    The *cpio* command is:

    ```
    cpio -itvh1
    ```

If this command fails at some point, the backup tape may be bad. Try to make the backup tape again, and verify the table of contents. If you cannot back up your disk, call the Geometry Hotline for assistance.

3. Become the super-user by typing:

    ```
    su
    ```

4. Shut down your system with the command:

    ```
    reboot
    ```

5. Put the GL2-W2.4 update tape in the tape drive.

6. Boot the *fex* program:

    IRIS 2400:

    ```
    b mt:dist/mdfex
    ```

    IRIS 2500:

    ```
    b mt:dist/ipfex
    ```

CAUTION
You must follow all instructions for *fex* commands precisely. Failure to follow these instructions can result in serious damage to your file system.

    The *mdfex* or *ipfex* initial screen and prompt appear:

    ```
    SGI Formatter/Exerciser Version 7.1 : March 1986
    Drive: 0, Name: name, Serial: serial number
    mdfex 7.1 >
    ```

7. Use the tape-to-disk command to copy the bootstrap tape to partition *b* on the disk. Initiate the process with this command:

    ```
    t
    ```

The system responds with the following prompt:

```
Tape to Disk copy
tape file (2)?
```

Press RETURN. The system responds with the following prompt:

```
Unit (0) =
```

Press RETURN. The system responds with the following prompt:

```
File System (a)?
```

Specify file system b instead by typing:

```
b
```

and press RETURN. The system responds with the following message:

```
copying 6000 blks in 500 chunks from tape file 2 to md0b
Type 'go < return>' to begin ...
```

Type:

```
go
```

and press RETURN. The system displays a series of messages as it copies the tape:

```
Started
copy started
1 2 3 4 5 6 7 8 9 10 11 12
Rewinding ...
Tape to disk copy complete
```

The *mdfex* prompt appears for the *IRIS 2400*:

```
mdfex 2.4>
```

The *ipfex* prompt appears for the *IRIS 2500*:

```
ipfex 2.4>
```

Type:

```
q
```

to quit. The system responds with the following prompt:

```
Quit - confirm quit with 'y':
```

Confirm quitting by typing:

```
y
```

This brings you back to the PROM monitor.

8. Boot the system you just read into the *b* partition by typing:

    IRIS 2400:

    ```
    b md0b:vmunix
    ```

    IRIS 2500:

    ```
    b ip0b:vmunix
    ```

    The system automatically converts the file systems to EFS, loads in the new standard software from tape, and reboots the system, returning you to the PROM monitor.

9. Bring the system back up by typing:

    ```
    b
    ```

    Make site-specific modifications to system configuration files. If you are in an Ethernet environment, edit the */etc/sys_id* file to include your system's hostname.

10. Type:

    ```
    multi
    ```

    to bring up the system in multi-user mode. The system prompts you for your model number. Enter one of the currently valid model numbers displayed and press RETURN . When the system asks:

    ```
    Do you want to check filesystem consistency? (y or n)
    ```

type:

```
y
```

11. If you have not purchased any software options, you have completed the software update. Go to step 22. If you have purchased software options, continue to follow the steps below.

12. Log in as root. Change to the root directory:

```
cd /
```

13. Read in the software distribution tools from the update tape:

```
cpio -ivhmud1
```

This command creates the distribution directory */dist* and reads in the set of commands and files that will be used during the rest of this procedure.

14. Change to the distribution directory:

```
cd /dist
```

15. Examine the table of contents in the update directory to determine the names of the options that are available by issuing the command:

```
cat toc
```

Each line of *toc* gives the name of an image on the tape and its length on the tape. The update tape consists of the standard images *boot*, *root*, and *usr*, followed by the software options you ordered. The *boot*, *root*, and *usr* images were already loaded during the update procedure above. Jot down the list of shorthand option names as they appear in *toc*, in the same order. You will supply this list of names as arguments to the *Read* and *Install* commands that follow.

16. Read in the software options with the *Read* command, specifying the shorthand names of the options as given in the *toc* file. For example, to read in the FORTRAN language option and the TCP/IP communications option, issue the command:

```
/dist/Read ftn tcp
```

17. Run the *Install* command to perform miscellaneous housekeeping functions, supplying the list of names that you specified for the *Read* command above. For example:

```
/dist/Install ftn tcp
```

As the *Install* command runs, it reports the names of the configuration files that are being updated. Examine each of these to determine if any site-specific modifications are required. See Section 4.2.4 in the *IRIS Series 2000 Owner's Guide, Version 2.0* for the complete list of configuration files. The *Install* command may ask you to run an `Install -cleanup` after you reboot the system, particularly if you have installed a communications option. If this happens, make a note of it.

18. Verify that certain critical files (listed in Table 4-4 of the *IRIS Series 2000 Owner's Guide, Version 2.0*) are still present:

```
/dist/Verify
```

This command does not perform a complete check of system integrity, but it can discover some of the problems that are likely to occur during installation of new software. Normally, *Verify* reports that there are no problems, and that it is OK to reboot the system. If problems are reported, however, correct them before rebooting, or it may be impossible to bring the system back up again. If you need assistance, call the Geometry Hotline.

19. If a communications option was installed, TCP/IP, IBM, or IEEE 488, the kernel included with the option will have been installed as */defaultboot* and */vmunix*. However, the system will still be executing the standard kernel. Reboot with the new kernel:

```
reboot
b
```

NOTE: to select explicitly a kernel to be installed as */defaultboot* and */vmunix*, use the *kernel*(1M) command.

20. Bring up the system in multi-user mode.

21. If *Install* asked you to run *cleanup* in step 17, type:

```
/dist/Install -cleanup
```

22. Remove the distribution directory:

```
rm -rf /dist
```

Version 1.0

Restore your user files from the tape you made in step 1 using this command:

```
tar xv
```

23. If you have TCP/IP, in the *Preparation* Section of this document, you were asked to save */etc/hosts* and */etc/hosts.equiv*. After the *Install* command is finished, use the *diff* command to compare old files to the new ones. For example, to compare */etc/hosts.old* to */etc/hosts*, issue these commands:

```
cd /etc
diff hosts.old to hosts
```

Make site-specific changes to these configuration files:

> */etc/hosts*
> */etc/hosts.equiv*
> */etc/rc.tcp*

If you have the IEEE 488 option, make site-specific changes to the configuration files */etc/rc.488* and */etc/ibtab*.

Do not substitute any old communication configuration files for the new ones; the new files are compatible with the update. See Section 4.4.2 of the *IRIS Workstation Guide, Series 2000* for a list of other configuration files.

NOTE: After you have installed the GL2-W2.4 release, you can not change your swap partition size.

4.4 Installing Updates over the Network

The following instructions apply to systems without a tape drive on an Ethernet network running XNS.

1. During this procedure, the entire contents of the disk will be destroyed and replaced with the new release of the standard system and software options. All files currently on the disk will be lost. Back up all user files that you want to keep, using *tar*(1) or *cpio*(1) to write them onto tape. Restrict the backup tape to user files only. Exclude all standard system files and directories supplied by SGI (see *Preparation*, above).

2. Perform a "table of contents" operation on the backup tape to ensure that it is readable. Both *tar* and *cpio* have a **t** option that does this. The *tar* command is:

    ```
    tar tv
    ```

    The *cpio* command is:

    ```
    cpio -itvh1
    ```

    If this command fails at some point, the backup tape may be bad. Try to make the backup tape again, and verify the table of contents. If you cannot back up your disk, call the Geometry Hotline for assistance.

3. Become the super-user by typing:

    ```
    su
    ```

4. Shut down your system with the command:

    ```
    reboot
    ```

5. To perform the update on a workstation with no tape drive, load the update tape onto a workstation with a tape drive. We will refer to that machine's hostname as *machine*. Make sure that the system with the tape drive is in multi-user mode.

6. On the machine to receive the updated software, enter the following commands:

    ```
    cd /
    cp vmunix patch
    adb -w patch
    ?m c00400 cfffff
    swplo?W 0d6000
    q
    reboot
    ```

7. The PROM monitor prompt appears. Issue this command:

    ```
    b patch
    ```

    This boots the system with a modified kernel. Once the system is up, don't enter multi-user mode. Issue these commands:

IRIS 2400:

```
xx machine mt rewind
xx machine mt fsf
xx machine dd if=/dev/rmt2 ibs=250k count=12 | dd obs=250k
of=/dev/rmd0b
reboot
```

IRIS 2500:

```
xx machine mt rewind
xx machine mt fsf
xx machine dd if=/dev/rmt2 ibs=250k count=12 | dd obs=250k of=/dev/rip0b
reboot
```

8. Boot the system you just read by typing:

    IRIS 2400:

    ```
    b md0b:vmunix
    ```

    IRIS 2500:

    ```
    b ip0b: vmunix
    ```

    Since the system cannot access the tape, you see this message:

    ```
    Can't access tape drive; assuming network update.
    Enter hostname of machine with tape drive:
    ```

    Type the hostname of the remote machine *machine*. The system automatically converts the file systems to EFS, loads in the new standard software from tape, and reboots the system, returning you to the PROM monitor.

9. Bring the system back up:

    ```
    b
    ```

10. If you have not purchased any software options, you have completed the software update. Go to the last step. If you have purchased software options, continue to follow the steps below.

11. Type:

    ```
    multi
    ```

    to bring up the system in multi-user mode. The system prompts you for your model number. Enter one of the currently valid model numbers displayed and press $\boxed{\text{RETURN}}$. When the system asks

    ```
    Do you want to check filesystem consistency? (y or n)
    ```

    type:

    ```
    y
    ```

12. Log in as root. Change to the root directory:

    ```
    cd /
    ```

13. To create the distribution directory */dist* and read in the set of commands and files that will be used during the rest of this procedure, enter this command:

    ```
    xx machine dd if=/dev/rmt1 bs=250k | cpio -ivhmud
    ```

    This command creates the distribution directory */dist* and reads in the set of commands and files that will be used during the rest of this procedure.

14. Change to the distribution directory

    ```
    cd /dist
    ```

15. Examine the table of contents in the update directory to determine the names of the options that are available.

    ```
    cat toc
    ```

    Each line of *toc* gives the name of an image and its length on the tape. The update tape consists of the standard images *boot*, *root*, and *usr*, followed by the software options you ordered. The *boot*, *root*, and *usr* images were already loaded during the update procedure above. Jot down the list of shorthand option names as they appear in *toc*, in the same order. You will supply this list of names as arguments to the *Read* and *Install* commands that follow.

Version 1.0

16. Read in the software options with the *Read* command, specifying the shorthand names of the options as given in the *toc* file. For example, to read in the FORTRAN language option and the TCP/IP communications option, issue the command:

    ```
    /dist/Read -x machine ftn tcp
    ```

17. Run the *Install* command to perform miscellaneous housekeeping functions, supplying the list of names that you specified for the *Read* command above. For example:

    ```
    /dist/Install ftn tcp
    ```

    As the *Install* command runs, it reports the names of the configuration files that are being updated. The *Install* command may ask you to run an `Install -cleanup` after you reboot the system, particularly if you have installed a communications option. If this happens, make a note of it.

18. Restore the configuration files that were saved on the backup tape.

19. Verify that certain critical files (listed in Table 4-4 of the *IRIS Series 2000 Owner's Guide, Version 2.0*) are still present:

    ```
    /dist/Verify
    ```

    This command does not perform a complete check of system integrity, but it can discover some of the problems that are likely to occur during installation of new software. Normally, *Verify* reports that there are no problems, and that it is OK to reboot the system. If problems are reported, however, correct them before rebooting, or it may be impossible to bring the system back up again. If you need assistance, call the Geometry Hotline.

20. If a communications option was installed, the kernel included with the option will have been installed as */defaultboot* and */vmunix*. However, the system will still be executing the standard kernel. Reboot with the new kernel:

    ```
    reboot
    b
    ```

    NOTE: to explicitly select a kernel to be installed as */defaultboot* and */vmunix*, use the *kernel*(1M) command.

21. Bring up the system in multi-user mode.

22. If *Install* asked you to run a *cleanup*, type:

    ```
    /dist/Install -cleanup
    ```

23. Remove the distribution directory:

    ```
    rm -rf /dist
    ```

24. Make site-specific modifications to system configuration files. Edit the */etc/sys_id* file to include your system's hostname.

25. If you have TCP/IP, in the *Preparation* Section of this document, you were asked to save */etc/hosts* and */etc/hosts.equiv*. After the *Install* command is finished, use the *diff* command to compare old files to the new ones. For example, to compare */etc/hosts.old* to */etc/hosts*, issue these commands:

    ```
    cd /etc
    diff hosts.old to hosts
    ```

    Make site-specific changes to these configuration files:

    > */etc/hosts*
    > */etc/hosts.equiv*
    > */etc/rc.tcp*

    If you have the IEEE 488 option, make site-specific changes to the configuration files */etc/rc.488* and */etc/ibtab*.

    Do not substitute any old communication configuration files for the new ones; the new files are compatible with the update.

26. Restore your user files from the tape you made in step 1. Make any necessary site-specific changes to the system configuration files. See Section 4.2.4 in the *IRIS Series 2000 Owner's Guide, Version 2.0.* for the complete list of configuration files.

    NOTE: After you have installed the GL2-W2.4 release,
    you can not change your swap partition size.

5. Additions

The next three sections describe additions to graphics, systems, and options software.

5.1 Graphics Additions

This section lists Graphics Library features added since the GL2-W2.3 release.

Item	Description
`blanktime()`	A new function `blanktime(nframes)` sets the duration before the screen is blanked. If *nframes* is 0, the screen never unblanks.
`getdev()`	A new command `getdev()` allows more than one valuator to be sampled in a single call.
`getothermonitor()`	The new function `getothermonitor()` returns the display mode available and not currently used.
`ismex()`	The `ismex()` routine returns TRUE if the window manager, *mex*, is running.
`isqueued()`	A new function `isqueued(device)` returns TRUE if the specified device is queued.
loadfont	The new program */usr/bin/loadfont* allows the default font to be set on system startup.
pop-up menu	A simple pop-up menu package allows user processes to use the same pop-up bit planes that the window manager uses. The 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 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 its 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()`	moves the lower left corner 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.

`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.

Version 1.0

vadvise	The *vadvise*() system call from 4.2BSD UNIX was added.

5.3 Additions to Options

This section lists additions to optional peripherals, communications options, and FORTRAN and Pascal tools since the GL2-W2.3 release.

Item	Description
dial and button boxes	Dial and button boxes and digitizer tablets can now be connected to ports 2, 3, and 4. A new command, *devport*, and a new Graphics Library routine, `devport`(), were 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.
light pen	A light pen option was added. A device, LPENVALID, acts like a button to indicate when the light pen position is valid.
Pascal	A Pascal local Graphics Library is now implemented.
TCP/IP	A new TCP/IP utility, *route*, provides host system access to the on-board routing table. This permits the system administrator to install routes through gateways on Ethernet, to systems on other networks.
	A new TCP/IP utility, *arp*, provides host system access to the on-board ARP table. The system administrator can examine the table, and can add or delete entries.
	A new TCP/IP daemon, *ud*(1M), extends *uucp* services (including mail) across the Internet protocols.
	The 4.2BSD standard TCP/IP *rwho* and *ruptime* utilities and their corresponding daemon, *rwhod*, were added.
wsiris	*wsiris* has a new keyboard command: *~DEL*. This reboots the IRIS after asking for confirmation.

Version 1.0

6. Changes

6.1 Graphics Changes

This section lists Graphics Library changes since the GL2-W2.3 release.

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	*/usr/include/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)	The cursor shape now belongs to the process that currently has the input focus. The menu demonstration program was removed. 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 a `winopen`() command.

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 (from other buttons and valuators) are delivered to the window that currently has the input focus, with the exception of the menu button. Pressing the menu button has two effects:

- 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 handled to this window. Applications that provide pop-up menus use this feature.

- Otherwise, the button-down event will go 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 allows that window to be moved, pushed or popped, etc. Pressing the menu button in the header of any window at any time 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 while using *mex*, examine your *.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.
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.
shaded polygons	Shaded polygons paint faster than in the 2.3 release.

6.2 Systems Changes

This section lists changes to UNIX since the GL2-W2.3 release.

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

Item	Description
as(1)	This new version of the assembler is the same as the version used on 68020 systems. The new assembler is faster than the old one, and will accept input from *stdin*. The old 68010

assembler is still available in */bin/oas*, for this release only. Besides extensions to the 68020 instruction set, the input languages for the old and new assemblers differ in these ways:

- The new assembler allows you to specify multiple instructions on a line by separating the instructions with a semicolon (;). The old assembler treated the semicolon as an undocumented secondary comment character.

- *as* accepts only the operators + and -. The old assembler recognized complement and multiply operators as well.

- *as* does not allow expressions to be placed in parentheses.

- *as* uses the syntax *:w* to specify word length in displacements and indexed expressions. The old assembler used the syntax *.w*.

- Some single-form instructions on the 68010 have multiple forms on the 68020. For consistency, some 68010 mnemonics changed. For example, the 68010 had only *chk*; the 68020 has *chkw* and *chkl*.

- *as* cannot generate assembly listings.

bcopy(), *bcmp*(), *bzero*()	The routines *bcopy*(), *bcmp*(), and *bzero*() were removed from */usr/lib/libxns.a*, because they are in */lib/libc.a*.
ci(1)	The length of the log message in RCS files was increased from 512 to 4096 characters.
ci(1), *co*(1)	A new option, **-M**[*rev*], causes the file's last-modified date, rather than the current date, to be used as the revision date. If the **–l** or **–u** option is also given, the file's last-modified date is not changed.
cp(1)	This command now has a **–r** flag for recursive tree copying.
cp(1), *mv*(1)	The 4.2BSD **–i** option was added. This option causes the system to prompt the user if *cp* or *mv* would overwrite an existing file.

Version 1.0

cpio(1)	When an error occurs, the system displays a description of the error rather than the value of *errno*.
dircmp(1)	A **−t** option has been added. It instructs *dircmp* to perform *diffs* only on text files.
/etc/bcheckrc	This file now asks if the file system consistency should be checked. *fsck* used to be in *root*'s *.login* or *.profile*, and now runs automatically.
/etc/checklist	This file is now created by */etc/brc* to contain device names appropriate to the model of machine.
/etc/inittab	The line in */etc/inittab* which executes */etc/brc* now allows input from the console.
/etc/model	This new file contains the model number of your IRIS system. If the */etc/model* file is missing or becomes corrupted, the system prompts for the model number during the transition from single- to multi-user mode. The response is written to this file.
/etc/passwd	The user *iris* was removed.
/etc/rc.xns	*xnsd* now starts the boot server *sgboot*.
error messages	The system now prints certain user-specific error messages on the user's console as well as on the standard console window.
fex	The *fex* program version number is 2.4.
floating point division	Software floating point division has been made more accurate.
kernel	The system does a better job of surviving swap errors. The message:

```
panic: swkill
```

was removed and replaced by:

```
sorry, pid n was killed due to no swap space
```

and

```
sorry, pid n was killed on swap error
```

Version 1.0

	The system now prints certain user-specific error messages on the user's console as well as on the standard console window.
/.login	This file no longer controls file system consistency checking.
ls(1)	*file* `not found` is now written to *stderr*, not *stdout*.
multi-user mode	Changes to the configuration file */etc/inittab* make it possible to boot into multi-user mode automatically. To boot into multi-user mode, modify the first line of */etc/inittab* file to read:

```
is:2:initdefault:
```

make	The command *make -p* now sends all of its output to *stdout* instead of sending some to *stdout* and some to *stderr*.
/.profile	This file no longer controls file system consistency checking.
reboot(1)	Many changes were made to the *reboot* procedure and programs (*/etc/init* and */etc/shutdown*). When you issue the *reboot* command, the kernel now kills all running processes by default, closes all open files, and unmounts all mounted file systems. The messages printed by *reboot* just before rebooting are immediately cleared. These messages are needed only if the reboot fails; they remain on the screen in that case.
	reboot asks for verification if it is being executed from a network tty.
reset(1)	This command now defines the terminal interface as in */usr/include/termio.h*. For example, the EOF character is now `CTRL-D` instead of `CTRL-A`.
/stand/mdfex, /stand/ipfex	The default copy-tape-to-disk partition is now **a** instead of **b**.
smt(1)	This command, used to give commands to a streaming magnetic tape drive, was replaced by *mt*.
/usr/include/termio.h	This file now includes a B19200 define for 19.2K baud.

Version 1.0

vi(1)   The maximum line length was increased from 1024 to 8192.

6.3 Changes to Options

This section lists changes to FORTRAN and Pascal tools and to communications options since the GL2-W2.3 release.

Item	Description
/etc/inittab	The **ttyp**[0-3] devices in */etc/inittab* have been changed to **ttyT**[0-8].
FORTRAN	*/usr/include/fgl.h* now contains a NAMEST parameter that tells the depth of the name stack.
	dbg, the debugger for FORTRAN, was removed from the release and replaced by *dbx*.
	f77 now includes a $SAVE option to add a SAVE statement at the beginning of each routine.
	The FORTRAN compiler *f77* now produces *.o* files rather than *.j* files.
	ar replaces the *library* command used with the FORTRAN archive.
IBM	Users can now escape *t3279* to a Bourne shell.
	mT3279 was removed, since its functions were incorporated into *t3279*.
IEEE 488	The IEEE 488 option has been made more robust.
Pascal	You now have more control over floating-point exceptions in Pascal programs.
	dbg, the debugger for Pascal, was removed from the release and replaced by *dbx*.
	The Pascal compiler *pc* now produces *.o* files rather than *.j* files.
	ar replaces the *library* command used with the Pascal archive.

TCP/IP

/etc/rc.tcp no longer kills *ftpd* and *telnetd* immediately after they are started.

Routing is now supported. The TCP/IP protocol module dynamically adapts routing policy according to ICMP redirect messages.

The ARP implementation can now advertise the physical addresses of third-party systems. This facilitates communication with systems that do not support the ARP protocol.

telnet and *rlogin* daemons are implemented on-board via the *xty*(7) driver. This obsoletes the old server programs *rlogind* and *telnetd*.

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.

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*.

The TCP/IP header files in */usr/include/EXOS* have new names:

Old Name	New Name
net/if_uba.h	*(no longer exists)*
net/in.h	*netinet/in.h*
net/in_pcb.h	*netinet/in_pcb.h*
net/in_systm. h	*netinet/in_systm.h*
net/ip.h	*netinet/ip.h*
net/ip_icmp.h	*netinet/ipicmp.h*
net/ip_var.h	*netinet/ip_var.h*
net/misc.h	*exos/misc.h*
net/netdb.h	*netdb.h*
net/pup.h	*exos/pup.h*
net/tcp.h	*netinet/tcp.h*
net/tcp_debug.h	*netinet/tcp_debug.h*
net/tcp_fsm.h	*netinet/tcp_fsm.h*
net/tcp_seq. h	*netinet/tcp_seq.h*
net/tcp_timer. h	*netinet/tcp_timer.h*
net/tcp_var.h	*netinet/tcp_var.h*
net/tcpip.h	*netinet/tcpip.h*
net/udp.h	*netinet/udp.h*
net/udp_var.h	*netinet/udp_var.h*

7. Bug Fixes

The next three sections list the bug fixes in IRIS graphics, system, and optional software since the GL2-W2.3 release.

7.1 Graphics Bug Fixes

This section lists bug fixes in the Graphics Library since the GL2-W2.3 release.

Bug	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 an 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 code that loads the font RAM no longer wraps around and writes over other programs' font RAM.
	Font RAM no longer cause horizontal lines.
`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.
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.
`linewidth()`	A call of `linewidth()` without an argument does not generate wide lines.
mex(1G)	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.

microcode	Stack overflows no longer occur when character moves are made outside of a bounding box, causing the character position to be clipped. After nine such clipped positions occurred, some subsequent microcode commands, such as polygons, did not work.
`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 comer of the graphics port.
`shaderange()`	The lint library for *libgl.a* now contains the correct four parameters for `shaderange()`.

Version 1.0

	The C and FORTRAN parameter type declarations are now correct.
`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	%**M** now echoes the correct monitor type.
	wsiris now recognizes `gbegin()`, `getport()` and `winopen()` as graphics initialization routines.

7.2 System Bugs Fixes

This section lists bugs fixes to UNIX, the C compiler, XNS, and the disk and tape drives.

Bug	**Description**
awk(1)	*awk* now prints the correct values for variables with *printf*(**"%g"...**).
C routines	Recursive C routines which are declared to be class *fortran* can now be called from C.
csh(1)	Changing the predefined environment variables *USER*, *LOGNAME*, *TERM*, or *HOME* now causes a corresponding change to the shell variable of the same name in lower case.
	The built-in *echo* command no longer causes *csh* to exit when given the command

```
% echo `echo "[m]"`
```

	when the **[m]** file does not match.
cxref	*cxref* now processes any C file that includes */usr/include/stdio.h*.
dbx	The *print* command in *dbx* no longer prints the original value of a variable after its value changed within a *while* loop.
eqn	*eqn* now forms the "there exists" special character correctly.

/etc/termcap	The */etc/termcap* entries underline start and end sequences for IRIS systems are now correct.
find(1)	*find* now interprets a question mark (?) correctly and no longer prints a period (.) for the current directory.
ipfex	*ipfex* can now be used to restore the root file system on the IRIS 2500.
kernel	The system no longer panics if a *-1* is passed as a system address on a machine without an FPA board.

The kernel scheduler no longer charges idle time to the last running process.

The exit of a child process whose *SIG_IGN* is set no longer creates a zombie process.

The paging system no longer generates a

```
dup mem alloc error.
```

A paging bug exhibited by the statements

```
system("date");
ginit();
gexit();
```

is fixed.

lex(1)	*lex* no longer dumps core on short erroneous input.
make(1)	*make* now expands the pseudo-variable **$?** correctly.
man(1)	Several minor bugs in the *man* command are fixed.
mkf2c(1)	*mkf2c* now handles null output from *extcentry*.
mv(5)	*mv* now handles the command *mv xxx dir* properly when *xxx* is a file and a directory named *dir/xxx* already exists.
permuted index	The permuted index generation macros */usr/lib/tmac/tmac.ptx* are now in place.
phys(2)	You must be the super-user to use this system call.

Version 1.0

prof(1)	*prof* now works when the user's text size exceeds 0x40000 bytes or the number of total functions in the executable file exceeds 300.
ps(1)	*ps* no longer produces the error message:

```
ps: error locating command name for pid %d from
```

when it dumps core or cannot find user program arguments.

Network tty names are no longer truncated to two character in *ps* output.

`ps u` now reports %CPU correctly.

ranlib(1)	Under certain conditions, *ranlib* produced an invalid library and the message:

```
no string table or truncated file
```

rogue(6)	Checking the load average and exiting *rogue* now work correctly.
strip(1)	*strip* now strips multiple archive files.
tar(1)	*tar* commands that used the **f –** option (used *stdin*) no longer produce a segmentation violation.
tbl(1)	*tbl* no longer dumps core.
test -x	If the super-user issues *test -x*, the system no longer returns TRUE when no execute permissions are set.
trap	Catching signal 18 with *trap* in *sh* caused a core dump. Signal 18 can no longer be trapped.
/usr/lib/cpp	The **–p** option of */usr/lib/cpp* now handles nested #*includes* properly.
USIZE	As documented in *core*(4), USIZE is now defined in */usr/include/sys/param.h*.
uucp(1C)	*uucp* now works with system names shorter than six characters.
xx	Background jobs started over the net with *xx host 'prog &'* now always execute.

XNS   If you interrupt a program that is writing to an XNS connection, it no longer logs you out of the program reading the connection.

                                 XNS no longer hangs with the error message:

```
nx: no message buffers.
```

xlogin(1)   XON/XOFF no longer causes inappropriate delays when you log in remotely to a workstation.

7.3 Options

This section lists bug fixes to peripherals options, communications options and the FORTRAN and Pascal compilers.

Bug	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.
digitizer tablet	There is now support for the digitizer tablet from FORTRAN.
/etc/rc.tcp	*/etc/rc.tcp* no longer requires that the local host be first entry in */etc/hosts*.
	/etc/rc.tcp now reports when the *netload* failed.
floating point hardware	The following program now behaves correctly when compiled for floating point hardware:

```
main()
{
float *rf0, *rf1;
*rf0++ += *rf1;
}
```

f77   The *f77* code generator no longer causes incorrect code to be generated when returning double-precision results of functions called indirectly.

Version 1.0

FORTRAN	The FORTRAN version of `shaderange`() now has the correct number of parameters.
	FORTRAN now truncates sequential files correctly upon close, using the new system command *ftruncate*.
FORTRAN functions	Several intrinsic FORTRAN functions are now defined:
	• *getarg*()
	• *iargc*()
	• *ran*()
FORTRAN graphics commands	Entries for several missing FORTRAN graphics commands were added to the interface library *libfgl.a*:
	• `gflush`()
	• `ginit`()
	• `greset`()
	• `gexit`()
	• `shaderange`()
IBM link	*/dev/pxd*, required by the IBM link, is included in the link software.
IEEE 488	The IEEE 488 driver now handles odd-length writes correctly.
	When you use the IEEE 488 interface, sequences of short (one- or two- byte) transfers no longer fail randomly.
	The IEEE 488 option file */etc/rc.488* now calls */etc/iib* instead of */etc/gpib/iib*.
	When IEEE 488 is used for graphics and serial line for text, all of the child processes are killed after *wsiris* exits.
mkf2c(1) and *mkc2f*	*mkf2c* and *mkc2f* no longer produce a core dump on input from *stdin*.
tar(1)	When you use the half-inch tape drive, issuing two *tar* commands one right after the other no longer causes the second one to fail.

TCP/IP

TCP now follows TIME-WAIT specifications. In the past, TCP had not closed 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 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` >.

8. Known Bugs

This list contains known problems with this release, and, where possible, suggested work-arounds.

Item	Description
check(1)	*check* -1 [*rcsdir*] does not work.
ci(1)	*ci* will not let you interrupt out of a check-in when performing a check-in on more than one file.
dbx	*dbx* garbles output when debugging from a terminal connected to a serial port. This problem does not occur, however, if the serial terminal is connected through *xlogin* to a workstation running *dbx*.

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
                                      work-around 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,...)
```

fast immediate mode macros   When using the fast immediate mode macros,
                                      there   are   times   when   the   file
                                      /usr/include/gl2/glerror.h   should   also   be
                                      included. To be safe, add the line:

```
#include "gl2/glerror.h"
```

                                      when you use the fast intermediate mode
                                      macros.

font RAM   When large fonts are repeatedly defined and
                                      deleted, the space in the font RAM is
                                      sometimes not cleaned up correctly, and a font
                                      cannot be defined correctly, even though there
                                      ought to be enough space to hold it.

getmem()   getmem() sometimes returns the wrong value.

mex(1G)   The textcolor and pagecolor commands do not
                                      work correctly under *mex*. Sometimes the
                                      color of the wrong textport is changed, and
                                      sometimes the commands appear to have no
                                      effect.

lines   Wide lines do not work correctly in depth cue
                                      mode.

Pascal Graphics Library   The Graphics Library commands readRGB(),
                                      writeRGB(), capture() and rcapture() do not
                                      work in the Pascal version of the Graphics
                                      Library.

	A Pascal program cannot be used to generate pictures on the color printer.
phys(2)	This system call is not supported in this release. For a work-around, contact the Geometry Hotline.
picking	Picking sometimes fails when objects are nested to a depth of more than six or seven.
sqrt(3M)	This program prints a very large number when it should print zero.

```
extern float sqrt();
main() {
long float dddl = 1.0;
printf("%f0.sqrt(float((float)(dddl-dddl)));
}
```

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 work-around described for `depthcue`(), above.