
IRIS

By: Nuyens (Nuyens.pa@Xerox.Com)

Requires: COLOR and COLOROBJ

This document last edited on January 31, 1987.

INTRODUCTION

The LispUsers module IRIS is a collection of functions for using an IRIS (Integrated Raster Imaging System) with Interlisp-D. The IRIS is a sophisticated 3-d color graphics workstation produced by Silicon Graphics Incorporated. Much of its imaging system is implemented in special purpose VLSI, making it a very powerful graphics engine.

IRIS provides three separate ways to exploit the IRIS in Interlisp-D. The first is to use the IRIS as an Interlisp-D imagestream. In this capacity the IRIS serves as the output device for a program which needn't even know that it is outputting to the IRIS. Thus standard system utilities (such as GRAPHER, SKETCH, and TEDIT) can make use of the IRIS without change. The images produced by these system utilities can then be manipulated with the special abilities of the IRIS (3-d rotation, color updates, double buffering, object definitions, etc.)

The second mode of use for the IRIS is to directly access the IRIS graphics library. In this mode, Interlisp-D provides stubs for the functions in the IRIS graphics library. Thus, Interlisp-D provides the illusion of having the IRIS as a direct device like the standard screen. However, it is actually a separate computer accessible across the 10-MB Ethernet.

The third mode is to use the view controller to interactively view scenes created with either of the two modes. The view controller allows the user to create an object once (interactively or via program) but then view the object many times on the IRIS without recreating the object. This object downloading allows real-time rotation of the object displayed on the IRIS.

This package also implements boot service for the IRIS workstation. The D-machine supplies the boot program to the IRIS over the Ethernet, eliminating the need for a floppy-disk on the IRIS.

This package benefited substantially from comments and improvements by Stu Card, Michel Desmerais and Lennart Lovstrand.

FILE DESCRIPTION

The implementation includes the following files:

LOADIRIS	after loading this file, the user will be prompted to place the IRIS icon. Buttoning this icon will produce a menu including the choice "create loadup panel". Choosing this item will produce a panel which can load the IRIS files. Button the "Standard" entry in the loadup panel, then button on the small SGI logo next to "Load". Prior to being able to run any applications, the IRIS must be running the terminal program. This can be booted from the floppy on the IRIS or from the Lisp workstation. To use the workstation as a bootserver, the boot file for the IRIS terminal program (see msg below) must be copied to {DSK}irisbootfile (or a directory in the list IRISBOOTDIRECTORIES) on the Dandelion that is to be the bootserver. At that point, hit the reset button on the IRIS and type 'n' to the monitor prompt on the iris. That will initiate a net boot from the Dandelion.
IRISSTREAM	The file which contains the Interlisp-D imagestream definition for the IRIS. Thus, the standard Interlisp-D graphics facilities can be applied to an open IRIS imagestream. After this is loaded, images which are processed through the standard hardcopy menu can be sent to the IRIS. "Iris" will appear as a printer when the "to a printer" choice is made from the roll-off menu entry "Hardcopy".
IRISLIB	this file contains the stubs for the functions in the IRIS library. For instance, corresponding to the function circf (found in the IRIS documentation), there is a function IRIS.CIRCF . The arguments are as listed in the IRIS documentation with the addition of the argument SPPstream. This is the SPP connection opened to the IRIS. (If omitted, it defaults to the value of IRISCONN, which is set by OPEN.IRISCONN). Where an argument is a matrix, it is passed to the lisp function as a list of the rows, each row itself a list. When a library fn returns a matrix, the appropriate (usually floatp) matrix is passed in and the elements are set.
IOLIB	this file contains the communication primitives which are used by the fns on IRISLIB.
IRISNET	this file contains the network support for the iris. The IRIS must be on the same network as the D machine (the IRIS doesn't handle routing through the gateway properly). If the IRIS is being bootservered from the Lisp workstation, the boot server software will set IRISNSHOSTNUMBER automatically. (The diagnostic messages can be inhibited by setting \IRIS.VERBOSE to NIL. They are left in because they can be useful when initially getting the IRIS to boot).
IRISVIEW	This file contains the functions supporting the interactive viewing menu, together with the object definition facility it uses.
IRISDEMOFNS	This file contains an example 3-d function for the IRIS. It is a function called TETRA which draws a colored, 3-d recursive tetrahedron.
IRIS.TEDIT	This file.

GETTING STARTED

N.B. All variables and functions in this document are written as in the old Interlisp readtable. To type in the examples in this document, bring up an "Old-Interlisp" Exec. (available from the background menu by rolling off "exec" and then "Interlisp".)

-1) type (FILESLOAD LOADIRIS)

It will ask you to position the following (SGI) icon:



(it can be recreated later by typing (IRIS.CREATE.ICON) at the Interlisp executive).

0) Next type (FILESLOAD IRISSTREAM IRISVIEW)

1) Boot the IRIS

If you have a floppy drive for your IRIS, use it to boot the IRIS GL2 (XNS) Terminal program. (If your IRIS is a workstation, not a terminal, you need to run the WSIRIS program available from SGI).

If you do not etherboot, you must set the variable IRISNSHOSTNUMBER. (see variables section below.)

If you have no floppy, you are going to "etherboot" the IRIS. To begin ensure that IRISBOOTDIRECTORIES is set properly. (see Magic Variables section below).

Press a mouse button in the SGI icon. Choose "Enable bootserver". Now boot the IRIS (by pressing the black button labelled "reset" on the IRIS display). To the prompt on the IRIS, type (on the iris keyboard) N (followed by carriage return). To the eventual prompt "connect to which host?" type < (carriage return). Etherbooting will automatically set the variable IRISNSHOSTNUMBER. Status messages will be printed to the promptwindow. The IRIS will beep when it is finished booting.

2) Open an IRIS stream

Button in the IRIS stream and choose "Create IRISView panel". Confirm the command by clicking left button of the mouse. "Connected" should print on the IRIS screen, and then the screen should turn black and display axes. (**N.B.** If at any time the screen goes completely blank, touch any key on the IRIS keyboard. The IRIS has a dubious screen-saver "feature".)

Modes

The modes of use of the IRIS are as follows:

Interlisp-D display stream

In this mode, the IRIS is acting like a window on the Lisp machine. To see updates as they occur toggle the "Double Buffer" button on the IRISView Controller. It will respond "Single buffering". Then press the "2D-Home" button on the IRIS. To demonstrate this, try the use of commands like

```
(PROGN (DSPCOLOR 'BLUE \IRISSTREAM)
```

```
  (PRINTOUT \IRISSTREAM T "Let's here it for " )
```

```
(DSPCOLOR 'RED \IRISSTREAM)
```

```
(PRINTOUT \IRISSTREAM "COLOR." T)
```

```
(DSPCOLOR 'BLUE \IRISSTREAM))
```

NOTE that the first time a font family is defined it will download the font after loading the spline font definitions. This will take a few minutes.

```
(DRAWLINE 0 0 400 200 2 NIL \IRISSTREAM 'RED)
```

```
(FORCEOUTPUT IRISCONN)
```

Since the SPP stream buffers, sometimes it will be necessary to (FORCEOUTPUT IRISCONN). (There is a function simply called F that will do this for you.)

These simple instructions indicate the use of the IRIS as an Interlisp Device Independent Graphics (DIG) stream. See the IRM for further examples common to all DIG devices. For interactive graphics, the following sections contain other uses of the IRIS.

Output stream for Sketch and TEdit

TEdit and Sketch can make use of the above display stream facilities. To see an example color document, do the following. (If you don't want to see the sketch in the document, you may omit loading sketch and sketchcolor, since they are slow to load.)

```
(FILESLOAD COLOROBJ SKETCH SKETCHCOLOR)
```


```
(TEDIT 'IRIS-EG.TEDIT)
```

To print a TEdit (or sketch) to the IRIS

First time: Press "2D-Home". In the title bar of the window, use the right button to bring up the default window menu. Pick the roll-off item Hardcopy. Rolloff into "hardcopy to printer", then choose "Iris". Choose "yes" to the menu saying "Make this the default printer".

Later times: The IRIS will now be the default printer, so just press "2D-Home" and choose Hardcopy from the default window.

To use Sketch

From the background menu, choose "SKetch". Build a closed polygon (with the  in the sketch menu). Choose "change" in the sketch menu. Pick a control point of the polygon you just built. Pick "filling color" from the menu that appears. From the menu that appears, pick a filling color.

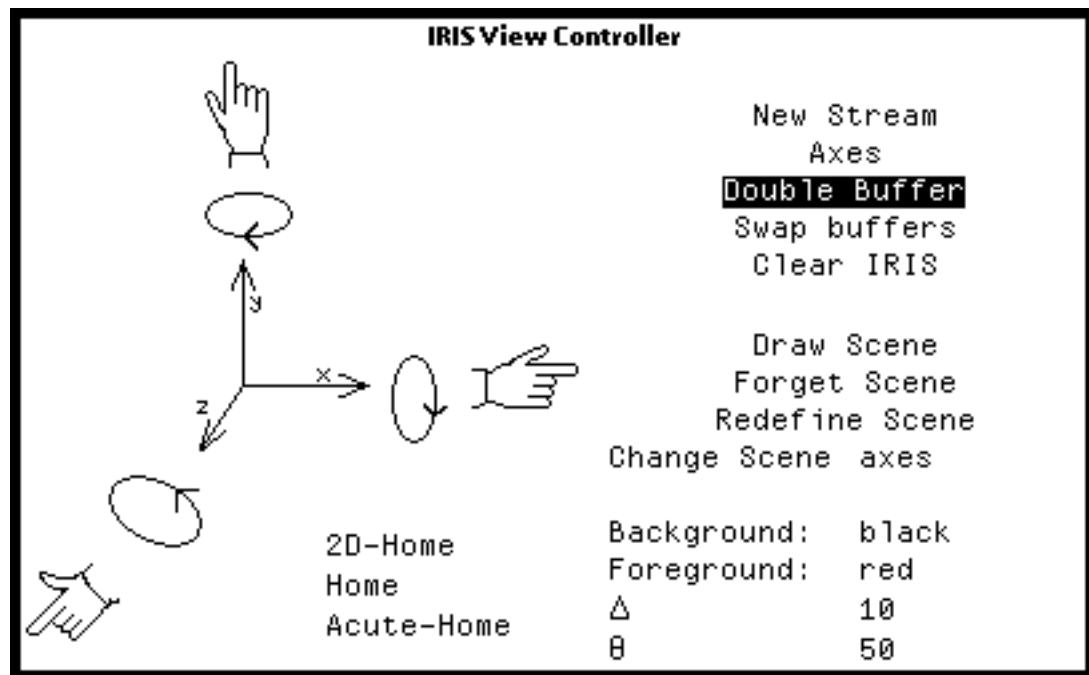
Clearing the IRIS

In the IRIS icon is an entry "clear IRIS". It will return the IRIS to the original state with the white screen, etc.

Defining new colors

At any time the color menu is presented, "new color" may be selected. RGB will provide color sliders and CNS will provide an english description of colors. A window will pop up asking for a name for the color. Type in a name, and that name will appear in the color menu from then on.

Using the IRISVIEW controller



N.B. When an IRISview controller is initially opened, the IRIS will be double buffering. This means that any drawing on the iris occurs on the back of two buffers. "Swap Buffers" must be chosen to bring the back buffer to the front. This includes "Clear IRIS" and other drawing commands. Double buffering can be turned off with the "Double Buffer" button, but moving images are much smoother with it turned on.

This window allows you to define objects on the IRIS and control your view of them.

Changing the scene:

First, select "change scene". You will be prompted with the predefined scenes as examples. Choose "axes". After a moment (during which "defining object for axes" is printed), an XYZ axes picture will appear on the screen. Try choosing each of the 3 home buttons to see the effect.

Press the "2d Home" button. (This actually means to move the view such that the IRIS simulates an 1108 screen in size and view). Now change the scene to "SKULL". Again after a delay, a Grateful Dead logo will be displayed.

Defining a scene:

We are going to define a scene that calls a function on the file IRISDEMOFNS so type

(FILESLOAD IRISDEMOFNS)

Button "Change Scene". Choose "New Scene" from the Menu. In response to "Scene Name", type "TETRA". In response to "Form to eval" click in the window the message is printed in and then type



(TETRA) [carriage return]

When the object is displayed, press each of the "Home" buttons to see different home positions. (2D-Home is perhaps best).

Viewing the scene:

Before viewing the scene, choose the "Double Buffer" option. Now hold the left button down on the item marked "Background:" in the view controller. Choose the background color for the object. White is often a good choice. (Note that sliding off the menu at this point will display "none" as the background. This means that each time the scene is drawn will be on top of the previous result. This can give very pleasing, flashy results. Experiment a bit. It's great fun (and you are probably being paid for it!))

Adjusting the view:

Bug the part of the free menu marked  This will move the image in the positive x axis. Similarly with the other hand icons. Holding down the shift key will reverse the direction in the same axis. Bugging the  entry in the view controller will let you change the "delta" for translation.



Now bug the part of the free menu marked . This will rotate the image in that axis. Holding down the shift key will reverse the direction of rotation.

Bugging the **B** entry will let you choose a new "theta" for rotation (in degrees).

Home:

Bugging "home" will return the scene to the original view. It is often a good idea to choose "Home" before changing scenes.

Axes:

Bugging the "axes" entry in the menu will superimpose a set of three dimensional axes on the view for reference purposes. It defines the axes of rotation controlled by the view controller and the directions of translation. (The axes look best against a black background, I think)

Forgetting a Scene:

Reclaims the space inside the IRIS and removes scene names from the scene menu. All but one scene may be deleted.

Defining a scene containing a sketch (or Tedit):

Button "Change Scene". Choose "New Scene" from the Menu. In response to "Scene Name", type "IRIS-EG". In response to "Form to eval", click in the window the message is printed in and then type a carriage return.

The view controller will prompt with "Make object then type RETURN". Now, hardcopy the sketch exactly as above. (periodically the view controller will flash the screen to remind you that you are still making the object). When you are finished, type carriage return as before. The current scene name will be printed in the view controller.

Troubleshooting:

"Iris Terminall SPP not responding" printing in prompt window

This may occasionally print when the Dlion is too busy to service the SPP connection to the IRIS. If it repeats, however, this means that a previous IRIS stream has been lost. Bring up a process status window by choosing PSW in the background, then choose the process (Iris Terminal SPP#2 for instance) and bug "kill" in the PSW. Be sure to kill the correctly numbered process. "Iris Terminal SPP" is not the same as "Iris Terminal SPP#3".

Magic Variables

IRISNSHOSTNUMBER

[Variable]

contains the 48 bit etherhostnumber of the IRIS. If you are etherbooting, this will be set automatically. Otherwise, set it to a string like "0#4000.12000.41504#0" as described on page 31.9 of the IRM.

\IRIS.DEBUG

[Variable]

Defaults to NIL. If T, when fonts are created, only the first lowercase letters will be defined. This is much faster than loading the whole font.

IRISBOOTDIRECTORIES

[Variable]

For users still running with the R1B version of the IRIS terminal program, the file concerned is called xiris. This is the file that should be copied to {core}irisbootfile. However, for users running R1C (also referred to as GL2), it is necessary to obtain an updated version of the terminal program from SGI. (It was named simply "iris" on the tape we received.) This variable must contain the list of directories where the boot files are to be found. (For instance, very fast booting may be obtained by copying the boot files to the {core} device and putting {core} at the front of IRIS.BOOTDIRECTORIES.)

IRIS.VERBOSE

[Variable]

Defaults to T. Says whether or not status messages are printed during font creation etc.

\IRISSTREAM

[Variable]

contains the IRIS stream that is current.

IRISCONN

[Variable]

contains the SPP connection that is current.