10 Customization

This chapter describes various ways of customizing SciAn to work well in your environment.

Speeding SciAn up

SciAn is designed to work on workstations with a wide range of computational and graphics capabilities. Settings which work fine when using small datasets on very fast workstations may not be adequate for larger datasets on slower workstations. This section describes a variety of ways to speed up the performance of SciAn.

Interaction

If rotating and moving visualizations within spaces is too slow, you can switch to a faster scheme which draws objects in a simpler way while interacting. See "Setting preferences" later in this chapter.

If you hold down the Alt key while moving a control, changes will be limited to the window you have pressed in until you release the key. This can be useful if the control affects the appearance of an object in another window, such as the brightness control of a visualization object.

If a visualization in a space takes too long to redraw and the delay is annoying, you can temporarily turn off the visualization object by selecting its icon and choosing Turn Off from the Object menu. If you hold down the Alt key while pressing the Visualize button or dragging into a visualization window's icon corral, the visualization object will be turned off by default. This allows you to set the controls and then turn the visualization object back on.

Scripts

Scripts typically contain a bunch of code to set up the visualization and then some recording commands. If the setup code is taking too long to run, you can speed it up by enclosing it in a setup block like this:

```
begin setup
      ...all the lines to set up the visualization go here
end setup
```

Within the `setup` block, most windows will not be redrawn. When the `end setup` command is read, all the windows will be redrawn. Because most visualization objects are not calculated until they need to be drawn, this will delay nearly all of the work until the end of the setup.

A `setup` block may not contain a `recording` block and vice-versa. This ensures that while you are recording, all windows will be redrawn.

Another way to speed up scripts is to include a `turn show controls off` line near the beginning of the script. This will prevent control panels from being shown in windows on the screen.

Setting preferences

There are several global preferences which can be used to customize your version of SciAn for your work. The preferences are set through a single window which is brought up by choosing the Preferences item from the main menu. Figure 10-1 shows a typical window; the window you see may not have all the items shown here.

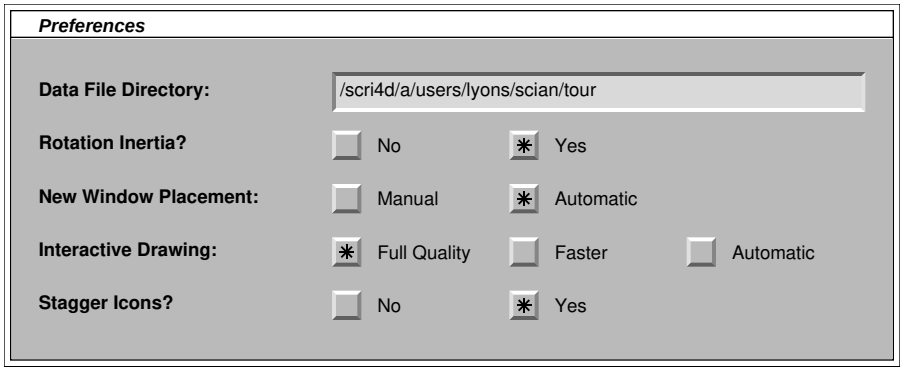

Figure 10-1. *Preferences window*

The Data File Directory text box contains the default directory to open when a new file browser is created. At first, this text box contains a single period (`.`) to indicate the current directory. You can change this to another directory to open by default. SciAn recognizes the tilde (~) to indicate your home directory. When the tilde is followed by a name, SciAn uses the home directory of the user given by the name.

The Rotation Inertia radio button group determines whether a space under rotation has inertia. If it is set to Yes, when you rotate a space and release the button while moving the mouse, the space will continue to rotate. If it is set to No, the space will stop when you release the button.

The New Window Placement radio group determines how windows are placed on the screen on the Silicon Graphics workstation. If Manual, you must click whenever a window is created to give it a location. If Automatic, the windows will automatically be placed on the screen in a staggered arrangement.

The Interactive Drawing radio button group determines how visualization objects are drawn when an interactive control is being used. If Full Quality is on, objects will always be drawn in full quality. If Faster is on, only the skeletons of objects will be

drawn to speed up interaction. If Automatic is on, SciAn will determine whether to use full quality or faster drawing depending on the complexity of the scene.

The Stagger Icons radio button group controls whether icons are staggered in corrals. If the button is selected, icons will be staggered up and down instead of placed in line side by side. This helps keep long icon names from overlapping.

Saving preferences

Preferences are automatically saved in a file called `.scianprefs` in your home directory when you exit SciAn normally. If you exit SciAn abnormally, such as by pressing Control-C on the console, any changes you made to the preferences in that session will not be saved.

Environment variables

SciAn is designed to adapt itself to the workstation on which it is running when it starts up. There may be cases where it is desirable to override some of SciAn's default assumptions, such as when SciAn must share the color table with other programs. This is done by means of environment variables which are set before SciAn is run.

Environment variables are set in most shells using the `setenv` command, which works like this:

`setenv` *variableName value*

When a command like this has been issued, the environment variable *variableName* is set with a value of *value*. The *value* is optional, and if it is not included, the environment variable is set with a blank value. This is sufficient for all but a couple of the variables recognized by SciAn. To unset an environment variable, use the `unsetenv` command like this:

`unsetenv` *variableName*

Some shells have an `env` command which can be used to set environment variables temporarily while a program is running. With some shells, the `setenv` command with no arguments lists all variables set. If you have any question about how this works on your system, consult the documentation or enter `man setenv`.

The following is a list of all environment variables recognized by SciAn:

`SCIAN_SHOW_CONFIG`
: If this variable is set, SciAn will print all the decisions it makes about the configuration of the system to standard output. This is a good first step in troubleshooting a problem with color usage and graphics.

SCIAN_VETO_DOUBLE
: Normally, SciAn will use double buffering if there are enough bit planes to do it. You can veto double buffering by setting this environment variable. Depending on what kind of workstation you are using, this may improve the quality of the images, but it will increase the amount of flashing on the screen.

SCIAN_VETO_RGB
: Setting this variable vetoes the use of full color (RGB) mode, even if your workstation appears to have enough bitplanes to use it.

SCIAN_VETO_CMAP
: Setting this variable vetoes the use of color map mode, even if your workstation appears to have enough bitplanes to use it. This is handy if you want SciAn to leave the color map entirely alone.

SCIAN_COLOR_BEG *colorBeg*
: When running on a system in color map mode, SciAn must allocate colors from the system's color table. Normally, SciAn will start with color 256 on a system with nine or more color map bitplanes. This works well with most systems; however, if you are running a program which requires values within that range, you may want SciAn to leave a certain range of colors alone. To do this, set the environment variable SCIAN_COLOR_BEG to the number of the color with which you want SciAn to start.

SCIAN_N_COLORS *nColors*
: Normally, SciAn allocates all colors from the beginning color (see SCIAN_COLOR_BEG) through the end of the color map. If you want SciAn to stop after a certain number of colors, you can set the SCIAN_N_COLORS variable to the number of colors to use.

SCIAN_VETO_RGB_GOOD_FOR_UI
: SciAn will use RGB mode for user interface windows by default, unless the number of bits for red, green, and blue are different. You can force SciAn to use color map mode by setting this environment variable.

SCIAN_FORCE_RGB_GOOD_FOR_UI
: If the number of bits for red, green, and blue are different, SciAn will not use RGB mode for user interface windows. This is because SciAn is not guaranteed to be able to use pure gray colors in this case. You can force SciAn to use RGB mode anyway by setting this variable.

SCIAN_RECORDER_DEV *portPath*
: The value of this environment variable gives the path name of the default serial port to use for communicating with the videodisc recorder driver. Use the complete path name (e.g. /dev/ttyz7). The device can be changed within the recorder driver control panel. The default recorder device for the LVR-5000 is /dev/ttyd4 and for the TQ-2026F is /dev/ttyd2.

References

Chen, M., Mountford, S. J., and Sellen, A., 1988: A Study in Interactive 3-D Rotation Using 2-D Control Devices. *Computer Graphics,* **22**(**4**), 121-129.

Cox, Donna J., 1988. Using the Supercomputer to Visualize Higher Dimensions: An Artist's Contribution to Scientific Visualization, *Leonardo*, **21(3)**, 233-242.

Foley, J. D. and Van Dam, A., 1982: *Fundamentals of Interactive Computer Graphics*, Addison-Wesley Publishing Company.

Kernighan, Brian W. and Ritchie, Dennis M., 1978: *The C Programming Language*. Prentice-Hall Inc.

Index

alpha blending 49
alpha planes 46, 69
alt key 3
alternate 3
angle of view 35
animation 79
  controllers 80
  dry run 90
  keyframe 87
  of time 64, 89
  of viewpoint 89
animations
  designing 87
  producing 88, 91
  with multiple windows 88
annotations 68
antialiasing 56, 57
arrows 43
backspace key 5
base color 47
baud rate 80
beveled boxes 69
bit planes 46
brightness 48
buttons 4
  cancel 8
  check boxes 4
  icon buttons 4
  OK 8
  plain 4
  radio buttons 4
  revert 21
cancel button 8
carpet plot 41, 55
check boxes 4
clicking 2
clipboard 5
clipping planes 35
clocks 62
  display 70
  running 64
  setting 63
  sharing 65
color
  bit planes 46
  color models 51
  components 51
  lookup table 46
  palette 30
  palettes 27, 49
  periodic functions 51
  ramp 50
  reverse 50
  ruffle 51
  saving palettes 52
  shading 48
  sharpen 50
  smooth 50
color bar 49
color cells 41
color combinations 87
color field 48
color map 46
color modes 46
color wheel 5, 50
communication controls 86
cones 54
conical frusta 54
constrain 3
contours 42
control key 3
controllers 34
controls 3
  color bar 49
  color wheels 5
  scroll bars 4
  sliders 4
  switches 6
  XY controls 5
copy text 5
corrals 6
creating visualizations 38
Crystal Eyes 36
current directory 21
current object 7
current window 8
cut text 5
cylinders 54
data 27
data forms 27
data modifiers 39
datasets 21, 27
  control panels 30
  main dataset 40
  modifying 31
  window 29
default directory 21
default visualization 38
deform 42
depth cueing 56, 57
device drivers 80
dialog windows 8
dimensions
  computational 28
  spatial 27
  topological 27
directory 22
dot files 23
dots 56
double-clicking 2
dragging 2
dragging icons 6
drawing objects 67
drawings 67
  annotation 68
  clock display 70
  depth 73
  lines 70
  moving 73
  palette legends 72
  rectangles 69
  reshaping 73
editing text 5
enclose values 40
enter key 5
exiting 10
extend 2
fields 29
file browser 21
file extension 26
file formats
  HDF 61
  STF 61
file readers 25
filename extension 22, 62
files

changing formats 25
data files 21, 24
formats 24
film recorders 79
fixed color 48
flight simulator 34
focus point 34
frame-by-frame recording 79
full color 46
full screen 87
gamma correction 47
geometry 27, 29, 44, 54
Gouraud shading 47
gray scale 52
grids 27
structured 28
unstructured 28
help 9
icon buttons 4
icon corrals 6
icons 6
dragging 6
selecting 6
image files 79
infinity 6
isosurfaces 40
isovalue 40
keyframe animation 87
keys
tab 68
laser videodisc recorders 80
lights 36
2-sided 53
brightness 37
color 37
number 36
position 37
shading 49, 53
line color 70
line width 69, 70
lines 55, 70
logs 21, 75
main dataset 40
maximum 42
menus 3
bringing up 3
submenus 3
mesh 55
meshes 41
minimum 42
missing data 6, 44
modifier keys 2
alternate 3
constrain 3
extend 2
shift 2
monitor overscan 88
mouse 1
center button 2
clicking 2
double-clicking 2
dragging 2
left button 1
pressing 2
right button 1
movies 79
NTSC 81
color choices 87
encoders 82
numbers 45
format 6
objects 1
current 7
selecting 7
shared 7
observer 33, 34
focus point 34
location 34
OK button 8
origin 42
orthographic view 35
overscan 88
PAL 81
palette 30
palettes 27
legends 72
Panasonic TQ-2026F 80
parent directory 23
parity 80
paste text 5
perspective view 35
Phong lighting 47
plain buttons 4
point clouds 43
pressing 2
printf 71
quitting 10
radio buttons 4
ramp 50
random jitter 44
rank 28
real time 62
record 90
recorder drivers 79, 84
recording block 89
rectangles 69
refining visualizations 46
renderer 37, 82
return key 5
reverse 50
revert button 21
RGB 46, 81
image files 79
rotate 33
rotation inertia 33, 89
RS-170A 81, 84
RS-232 83
ruffle 51
S-VHS 81
saving palettes 52
scan converters 82
scripts 21, 75, 88
scroll bars 4
selecting icons 6
selecting objects 7
selection rectangle 6
serial port 80, 83, 86
setup block 90
shadow 69
shared objects 7
sharing clocks 65
sharpen 50
shift key 2
sliders 4
smooth 50
snap 90
Sony CVR 80
Sony LVR-5000 80
space panels 32, 66
space time 62
spaces 32
controllers 33, 34
controls 34
panels 32
tools 32
specular highlights 52
spheres 54
stereo 35
stop bits 80
subdirectory 23
submenus 3
surfaces 52

switches 6, 48
templates 39
text
  copy 5
  cut 5
  editing 5
  paste 5
text annotation 68
text boxes 5
time
  animating 62, 64
  changing time 62
  clocks 62
  format 71
  real time 62
  space time 62
  video time 62
time dependence 29
time-dependent data 61
time-dependent files 61
timesteps 61
translucency 49, 53, 69
transparency 49, 53, 69
two-sided lighting 37
user interface 1
video boards 82
video recorder 79
  connecting 79
video screen 87
video systems 81
video time 62
visualization objects 32, 40
  attributes 40
  controls 40
visualization window 32
visualizations
  arrows 43
  carpet plot 41
  contours 42
  creating 38
  geometry 44
  isosurfaces 40
  meshes 41
  numbers 45
  point clouds 43
  refining 46
width of view 35
windows 1, 8
  current 8
  dialog 8
XY controls 5
Z-buffer 66