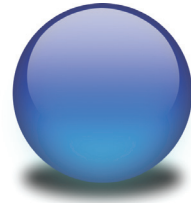


Index



Note: Information presented in tables and figures is denoted by t and f respectively.

A

- Activity Monitor, 227
- Advanced Graphics Port, 29, 30t
- AGL. *see* Apple OpenGL
- AGL.ACCELERATED, 99t
- AGL.ACCUM_ALPHA_SIZE, 97t
- AGL.ACCUM.BLUE.SIZE, 97t
- AGL.ACCUM.GREEN.SIZE, 97t
- AGL.ACCUM.RED.SIZE, 96t
- AGL.ALL_RENDERERS, 95t
- AGL.ALPHA_SIZE, 96t
- AGL.AUX_BUFFERS, 96t
- AGL.AUX_DEPTH_STENCIL, 98t
- AGL.BACKING_STORE, 100t
- AGL.BLUE_SIZE, 96t
- AGL.BUFFER_SIZE, 95t
- AGL.CLOSEST_POLICY, 97t
- AGL.COLOR_FLOAT, 98t
- AGL.DEPTH_SIZE, 96t
- AGL.DOUBLEBUFFER, 95t
- AGL.FULLSCREEN, 98t
- AGL.GREEN_SIZE, 96t
- AGL.LEVEL, 95t
- AGL.MAXIMUM_POLICY, 97t
- AGL.MULTISAMPLE, 98t
- AGL.MULTISCREEN, 100t
- AGL.NO_RECOVERY, 99t
- AGL.NONE, 101t
- AGL.OFFSCREEN, 98t
- AGL.PBUFFER, 100t
- AGL.PIXEL_SIZE, 97t
- AGL.RED_SIZE, 96t
- AGL.REMOTE_PBUFFER, 101t
- AGL.RENDERER_ID, 99t
- AGL.RGBA, 95t
- AGL.ROBUST, 100t
- AGL.SAMPLE_ALPHA, 99t
- AGL.SAMPLE_BUFFERS_ARB, 98t
- AGL.SAMPLES_ARB, 98t
- AGL.SINGLE_RENDERER, 99t
- AGL.STENCIL_SIZE, 96t
- AGL.STEREO, 95t
- AGL.SUPERSAMPLE, 98t
- AGL.VIRTUAL_SCREEN, 100t
- AGL.WINDOW, 100t
- aglChoosePixelFormat, 93–94, 95t–101t
- aglCreatePBuffer, 110t
- aglDescribePBuffer, 110t
- aglDestroyPBuffer, 110t
- aglGetPBuffer, 110t
- aglSetPBuffer, 110t
- aglTexImagePBuffer, 110t
- AGP. *see* Advanced Graphics Port
- alignment
 - pixel data, 224–25
 - texture, 225–26
- alternative rendering destinations, 109–13, 312–22
- Altivec engines, 28, 42–43
- API layers, 15–16, 16f
- APIs
 - and surfaces, 15–16
 - cross-platform, 48–49
 - integration with, 254–55
 - Mac-only, 46–49
 - X11, 277–79

-
- AppKit, 16, 16f, 20, 122–33, 123f, 124f, 125f, 126f
 - Apple Fence, 216–17
 - Apple Float Renderer, 13
 - Apple Generic Renderer, 13
 - Apple OpenGL, 16, 16f, 47
 - alternative rendering destinations in, 109–13
 - context sharing in, 107–9
 - framebuffer objects in, 117–19
 - full-screen application in, 91–101
 - pbuffers in, 110–13, 110t, 113f, 208t
 - renderers in, 104–7, 107t
 - software layering in, 90–91, 91f, 91t
 - windowed application in, 101–4
 - Apple Texture Range, 205–7
 - Apple Vertex Array Range, 205–7
 - APPLE_flush_buffer_range, 220–21
 - APPLE_vertex_array_range, 217–19
 - ARB. *see* Architecture Review Board (ARB)
 - Architecture Review Board (ARB)
 - and extensions, 8–9
 - creation of, 8
 - asynchronous calls, 204–7
 - B**
 - best practice axioms, 196–201
 - BGRA pixel format, 10t
 - bitmap, to NSImage, 183–84
 - blend squaring, 10t
 - blending logical operations, 9t
 - Breakpoints view, 229–31, 230f
 - buffer flush, 205–7, 220–21
 - buffer sizing, 59–60
 - Buffer view, 234, 235f
 - bugs, in OS, 39–41
 - bus
 - bandwidth, 30
 - graphics, 29–30, 30t
 - memory, 29
 - C**
 - C (programming language), 2, 89–90
 - C++, 89–90
 - cache, in CPU, 25
 - CAD limitations, 27
 - CAE limitations, 27
 - central processing unit (CPU)
 - and Activity Monitor, 227
 - and clock rate, 28
 - cache, 25
 - idle time minimization, 204–7
 - northbridge of, 23–24, 24f
 - southbridge of, 24, 24f
 - transistors in, 23
 - CGL. *see* Core OpenGL (CGL)
 - CGLChoosePixelFormat, 58–59
 - CGLFlushDrawable, 230, 231–32
 - CGLRendererInfoObj, 72
 - Cheetah. *see also* OS X
 - release of, 17t
 - clock rate, 28
 - Cocoa API in Leopard, 283–322
 - Cocoa Image, 174–84, 178f, 179t
 - Cocoa OpenGL, 47–48
 - color models, X11, 279
 - context enables, 64–65
 - context management, in CGL, 63–68
 - context parameters, 64–65
 - context sharing
 - in AGL, 107–9
 - in OpenGL, 141–49, 143f, 144f
 - in OpenGL Leopard, 301–8
 - copy-to-texture, 114–17, 158–61, 318–21
 - Core Graphics, 16, 49
 - Core OpenGL (CGL), 16, 16f, 18, 47, 53–54, 54f, 55–87
 - buffer sizing, 59–60
 - ChoosePixelFormat, 58–59
 - context management in, 63–68
 - error handling in, 57
 - global state functions in, 84–86
 - macros in, 86
 - pbuffer selection mode, 208t
 - pixel format selection in, 57–63
 - read-only parameters, 68
 - read/write parameters in, 66–67
 - virtual screen management in, 83–84
 - Core Video, 192
 - CoreFoundation, 37–38
 - CPU. *see* central processing unit (CPU)
 - D**
 - Darwin, 33
 - data copy minimization, 201–2
 - data flow
 - across contexts, 51–53, 76–77
 - and hardware, 24–32
 - unidirectional, 199–200
 - data management, 209–10

data parallel computation, 42
debugging, OpenGL, 39–41
dependency layering, 50, 50f
depth textures, 10t
development, history of Mac, 2–3
display capture, 149–51, 309–10
display IDs, 70–71
display masks, 20, 71
display release, 150
displays, 26–27
dl* calls, 265–66
dlopen, 265–66
dlsym, 265–66
double-buffered drawables, 77–78
downloadImageAsTexture, 188–89
drawables, 77–86
drawRect, 132–33
Driver Monitor, 235–36, 235f, 236f
driver plug-ins, 17–18
drivers, and renderers, 21

E

encapsulation, 197
error handling, in CGL, 57
event handling, 151–52, 310–12
EXT extensions, 8–9
extensions
 and ARB, 8–9
 creation of new, 8
 design of, 254–55
 EXT, 8–9
 identification, 257–62
 management libraries, 269–75
 overview of, 253–54
 parallel, 8
 promotion of, to Core OpenGL,
 9t–11t
 query, 257–62
 range, 205–7
 selection, 257–62
 styles, 256
 tokens and, 259–60, 260t
 types, 256
 usage, 257–62
 utilization and binding, 262–69

F

FBOs. *see* framebuffer objects
feature support, 14–15
fences, 207, 216–17
filesystem, 38

fill, 26
floating-point framebuffers, 23
Flush Buffer Range extension, 205–7,
 220–21
fragment shaders, 200–201, 243
frame rate metrics, 207–8
frame rate quantization, 26–27, 208
framebuffer objects, 117–19, 153–58,
 158f, 313–18
framebuffer-to-texture copy, 9t
framebuffers, 53–54, 54f
framework locations, 50t
full-screen application, in AGL, 91–101
full-screen drawables, 82–83
full-screen surfaces, 149, 308–9
function call overhead, 204

G

gaming limitations, 27
GL_EXTENSIONS, 249t
GL_RENDERER, 249t
GL_TRIANGLE_STRIP, 211
GL_VENDOR, 249t
GL_VERSION, 249t
GL_ARB_non_power_of_two, 176
GLEW, 270–73
glFinishFenceAPPLE, 217
global state functions, in CGL, 84–86
glReadPixels, 241–42, 242f
glTexImage2D, 197
GLUT, 16, 16f, 48
 and extension management, 273–75
 configuration of, 165–70
 history of, 163
 overview of, 164, 164f
 pixel format in, 167–70, 167f, 169t, 170t
 window management in, 169, 169t
GLUT_ACCUM, 169t
GLUT_ALPHA, 169t
GLUT_DEPTH, 169t
GLUT_DOUBLE, 169t
GLUT_MULTISAMPLE, 169t
GLUT_RGB, 169t
GLUT_RGBA, 169t
GLUT_SINGLE, 169t
GLUT_STENCIL, 169t
GLUT_STEREO, 169t
glutInitDisplayString, 170, 170t
glVertexArrayParameteriAPPLE, 216
GLX, 51

GPU. *see* graphics processing unit (GPU)
graphics bus, 29–30, 30t
graphics device support, 7
graphics processing unit (GPU)
 bandwidth limits with CPU, 25
 idle time minimization, 204–7
 increasing role of, 23
 information lookup, 34
 transistors in, 23
graphics tools, 228–36, 229f, 230f, 232f, 233f,
 234f, 235f, 236f, 237f

H

hardware
 and data flow, 24–32
 image download to, 180–82
 on Mac platform, 7
 renderer support, 12t–13t, 13–14
hardware *vs.* software renderers, 19
history, of Mac platform, 2

I

idle time minimization, 204–7
image manipulation, 140–41
imaging subset state, 223–24
immediate mode rendering, 198–200
immediate mode vertex submission, 211–12
implicit data copies, 29
instance method, 146–47
intermediate rendering, 152–53, 321–22
IOKit, 37–38
IrisGL, 3, 8

J

Jaguar. *see also* OS X
 improvements of, 246
 release of, 17t
 renderer support, 12t
 VRAM requirements in, 31

K

Kilgard, Mark, 163

L

Launch Settings menu, 228–29, 229f
Leopard. *see also* OS X
 Cocoa API on, 283–322
 improvements of, 248
 Quartz in, 31
 release of, 17t
 renderer support, 13t
 threaded engine on, 202

M

Mac platform
 advantages of, 3
 development on, 2–3
 hardware, 7
 history of, 2
 user experience, 2
Mach threads, 41
macros, in CGL, 86
medical imaging limitations, 27
memory
 requirements, 31–32
 video, 30–31
memory bandwidth limit, 25
memory bus, 29
metrics, 207–9
Movie Toolbox, 189–90
multisampling, 10t, 61
multithreaded engine, 202–4

N

NextStep, 2
northbridge, of CPU, 23–24, 24f
NSImage, 174–84, 178f, 179t
NSMovie, 185
NSOpenGLPFACcelerated, 131t, 292t
NSOpenGLPFACcumSize, 130t, 291t
NSOpenGLPFAlphaSize, 129t, 290t
NSOpenGLPFAAuxBuffers, 129t, 290t
NSOpenGLPFAAuxDepthStencil, 131t, 292t
NSOpenGLPFABackingStore, 131t, 292t
NSOpenGLPFAClosestPolicy, 131t, 292t
NSOpenGLPFAColorFloat, 131t, 292t
NSOpenGLPFAColorSize, 129t, 290t
NSOpenGLPFADepthSize, 130t, 291t
NSOpenGLPFADoubleBuffer, 129t, 290t
NSOpenGLPFAScreen, 130t, 291t
NSOpenGLPFAllRenderers, 129t, 290t
NSOpenGLPFAMaximumPolicy, 130t, 291t
NSOpenGLPFAMinimumPolicy, 130t, 291t
NSOpenGLPFAMultisample, 131t, 292t
NSOpenGLPFAOffScreen, 130t, 291t
NSOpenGLPFAPixelBuffer, 131t, 292t
NSOpenGLPFARendererID, 131t, 292t
NSOpenGLPFASampleBuffers, 130t, 291t
NSOpenGLPFASamples, 131t, 292t
NSOpenGLPFAStencilSize, 130t, 291t
NSOpenGLPFAStereo, 129t, 290t
NSOpenGLPFAStereo, 129t, 290t
NSOpenGLPFAStereo, 129t, 290t
NSOpenGLPFAStereo, 129t, 290t
NSOpenGLPFAStereo, 129t, 290t
NSOpenGLPixelFormat, 129t, 290t–131t,
 292t

NSOpenGLView, 122–33, 123f, 124f, 125f,
126f, 284–93

NSSlider, 221

NSView, 133–40, 134f, 135f, 294–300

O

object-oriented programming, and state
management, 197–98

Objective-C, 89, 121

off-screen drawables, 82

off-screen rendering, 161–62, 321–22

off-screen surfaces, 109–10

OpenGL

advantages of, 3–4

as specification document, 8

debugging, 39–41

feature support, 14–15

graphics tools, 228–36, 229f, 230f, 232f,
233f, 234f, 235f, 236f, 237f

history of, 7–8, 9f

platform identification, 248–49, 249t

OpenGL Driver Monitor, 235–36, 235f, 236f

OpenGL Profiler, 228–34, 229f, 230f, 232f,
233f, 234f, 235f

OS 9, 2

OS requirements, 27–28

OS version identification, 249–51

OS X. *see also* Cheetah; Jaguar; Leopard;
Panther; Puma

10.0 through 10.1, 245–46

filesystem, 38

implementation of OpenGL specification,
11

power management in, 34–38

tools, 226–28

version history, 17t, 33–34, 34t

P

Panther. *see also* OS X

improvements of, 246

QTKit in, 185

renderer support, 12t

parallel extensions, 8

Pascal (programming language), 2

pbuffers, 78–81, 79t, 110–13, 110t, 113f,
161–62, 208t, 321–22

PCI Express, 29–30, 30t

pixel buffer objects, 11t, 241–42

pixel buffers, 78–81, 79t, 110–13, 110t, 113f

pixel data, from NSImage, 178–79, 179t

pixel data alignment, 224–25

pixel data processing, 26

pixel format

and texture data handling, 221–23

in AGL, 91–109

in CGL, 57–63

in GLUT, 167–70, 167f, 169t, 170t

Pixel Format view, 234, 234f

pixel manipulation, 140–41

pixel pipeline, 223–24

pixel types, 221–23

Pixie, 235, 236f

Please Tune Me, 237–43, 239f, 240f, 241f,
242f, 243f

plug-in architecture, 17–18, 18f

point parameters, 10t

polygon offset, 9t

power management, 34–38

PowerPC, 42–43

Profiler, 228–34, 229f, 230f, 232f, 233f,
234f, 235f

programmable shading, 11t

ptm1.c, 238–39, 239f

ptm2.c, 240, 240f

ptm3.c, 240–41, 241f

ptm4.c, 241

ptm5.c, 241–42, 242f

Puma. *see also* OS X

release of, 17t

Q

QTKit, 184–88, 191–92

QTMovie, 186–87, 191–92

quad-buffering, 61

Quartz, 15, 16, 46–47

Quartz 2D, 31

Quartz Debug, 228

QuickTime, 184–92

R

range extensions, 205–7

read-only parameters, 68

read/write parameters, 66–67

render targets, 60–61

render-to-texture, 114–17, 158–61

renderer support, 12t–13t, 13–14

renderers, 18–21

and intermediate rendering, 152–53

choosing, 21

definition of, 18

drivers, 21

hardware *vs.* software, 19

in AGL, 104–7, 107t

incompatibility of, 141–42

renderers, (*continued*)
 obtaining information on, 68–76
 selection in CGL, 56f, 61–63
 switching between, 19–21
rendering, retained *vs.* immediate mode,
 198–200
rendering destinations, alternative, 152–62
Resources view, 233, 233f
retained mode rendering, 198–200

S

Scripts view, 234, 235f
secondary color, 10t
shader instruction limits, 201
shader performance, 226
shader use, 200–201
shadow functions, 11t
Shark, 3, 227
Single Instruction, Multiple Data (SIMD), 42
singleton, 142, 144, 146–47
software layering, in AGL, 90–91, 91f, 91t
software *vs.* hardware renderers, 19
southbridge, of CPU, 24, 24f
sRGB textures, 11t
state change minimization, 196–98
Statistics view, 231, 232f
stencil wrap, 10t
stereo rendering, 61
subset state, 223–24
subtexturing, 9t
surfaces, and windowing systems, 15–16
system tools, in OS X, 226–28

T

teardown, 150
texture
 alignment considerations, 225–26
 and data integrity, 176
 compressed, 225
texture data handling, 221–25
texture ID, 188–89
texture LOD bias, 11t
texture objects, 10t
texture proxying, 9t
texture range extension, 205–7
texture, NSIMage as, 179–82
texturing, with pbuffers, 81
threading, 41–42, 42f

throughput, 209
Tiger. *see also* OS X
 improvements of, 246–48
 UIKit in, 185
 RAM requirements, 31
 release of, 17t
 renderer support, 12t
tokens, and extensions, 259–60, 260t
Trace view, 231–32, 232f

U

unidirectional data flow, 199–200
Unix core, 33
user experience, on Mac, 2

V

VAR extension. *see* Vertex Array Range
 extension
VBOs. *see* vertex buffer objects (VBOs)
vendor extensions, 8
version information access, 14
vertex array objects, 214–15
Vertex Array Range extension, 215–20
vertex arrays, 10t, 205–7, 211–12
vertex buffer objects (VBOs), 212–15,
 220–21
vertex data handling, 210–21
vertex data processing, 26
vertex shaders, 200–201
vertex submission, 210–21
video editing limitations, 27
video memory, 30–31
virtual desktops, 19
virtual screen management, 83–84
VRAM. *see* video memory

W

window management, in GLUT, 169, 169t
windowed application, in AGL, 101–4
windowing systems
 consistency between, 15
 data structure of, 15–16

X

X11, 15, 49, 277–79
X11 APIs, 277–79
X11 color models, 279
XCode, 3