

Index

NUMERALS

3D listener object, 264
3D models, 119–121
 file formats, 63
 lines, rendering, 123–127
 MeshRenderManager, 416
 textures, rendering, 121–123
 unit testing, 131–135
 XnaShooters, 301–303
3D programming, 127–131

A

AddChatMessage method, 479
AddEffect method, 319
AddRandomBlock method, 104
AddSpriteToRender method, 122
Ageia PhysX, 381–382
Agine Methodology, 32–33
AI (Artificial Intelligence), Dungeon Quest, 530
AlphaBlending property, BaseGame, 126
animated textures, XnaShooter, 303–304
AnimatedModel class, 524
AnimatedModel classes, 523
AnimatedModel.cs, 296
AnimatedTexture, 304
annotations, 156
Ants Profiler, 24
ApplyGravity method, 378
ApplyGravityAndCheckForCollisions method, 395
ArrayList class, 76

Assert helper class, 71
Asteroid.cs, 296
Asteroids, 178–182
Automatic, VertexBuffers and, 213

B

background
 scrolling, 20
 Tetris, rendering, 100–102
 textures, displaying, 19
BallSpeedMultiplier, 46
BaseAsteroidManager.cs, 296
BaseGame class, 81, 307
 AlphaBlending property, 126
BaseMessage class, 458, 465
BasicEffect class, 124
BeginReceiveFrom method, 460
BeginSendTo method, 461
Big-Button-Pad, 269
billboards, XnaShooter, 305–308
Billboard class, 306
BlockColors, 104
BlockFalldown sound, Tetris, 99
BlockMove sound, Tetris, 99
BlockRotate sound, Tetris, 99
BlockTypes enum, 104
BlockTypeShapes, 104
.bmp file format, 63
bounding boxes, 51
Breakout game, 80–82
 collision testing, 88–89
 Draw method, 86–87, 86–88

Breakout game (continued)

Breakout game (continued)

drawing, 86–88
 game loop, 85–86
 improving, 89–90
 levels, 83–85
 unit testing, 82–83

BreakoutBlockKill.wav, 82

BreakoutGame.png, 82

BreakoutVictory.wav, 82

Bullet Physics Library, 384

C

C#, game development and, 22–23

CalcLandscapePos helper method, 344

CalculateTangentFrames, 196

CalculateTangentMatrix, 185

CalcVectors helper method, 307

Camera class, 137–139

Canyon Commander, 401–403

cars, more, 433
 concept, 403–404
 effects, 434–435
 final testing
 Windows, 428–430
 Xbox 360, 430–432
 game idea, 405–407
 game screen, logic, 407
 game screens, 409–410
 Highscores, 417–418
 main menu, 413–414
 splash screen, 410–412
 GameScreen class, 414–417
 graphics engine, 407
 Helper classes, 407
 highscores, online, 433–434
 modes, 435–436
 multiplayer mode, 436–437
 RenderToTexture class, 408
 shaders, 407, 434–435

shadow mapping classes, 422–428

ShadowMapBlur class, 408

ShadowMapShader class, 409

tweaking, 421–422

unit testing, 408, 418–420

write your own mode, 437–442

car collision, 395–399

car physics, 375

CarPhysics class, 373

ChaseCamera class, 378

ChatMessage, 458

Chatpad device, 269

CheckBallCollisions method, 86

classes

AnimatedModel, 524

ArrayList, 76

BaseGame, 81, 307

BaseMessage, 465

BasicEffect, 124

Billboard, 306

Camera, 137–139

CarPhysics, 373

ChaseCamera, 378

ColladaCombiModels, 356

ColladaLoader, 356

ColladaModelFile, 523

ColladaTrack, 356

DrawableGameComponent, 93

EditorScreen, 539

EffectManager, 295

EndGameMessage, 467

FontToRender, 97

Game, 92–93

GameServer, 456

Highscore, 433

Input, 267, 268–271

Item, 325–327

JoinGameMessage, 464

Landscape, 342

LevelGroundRender, 540

CreateScale method

- LineManager3D, 125
- LoadLevel, 343
- Log, 68
- Material, 190
- MeshHelper, 197
- Mission, 312
- Model, 132
- Multiplayer, 457
- NextBlock, 105
- NumbersFont, 282
- ParallaxMapping, 407
- PhysicsAsteroidManager, 392–395
 - player, 457
- PostScreenDarkenBorder, 221–224
- PostScreenGlow, 407
- Projectile, 323–325
- RandomHelper, 73–74
- RelativeScreenRect, 478
- RenderTarget, 219
- RenderToTexture, 220–221
- RocketCommanderGame, 296
- ScreenshotCapturer, 139
- ServerGame, 457
- SessionConnection, 456, 462
- ShaderEffect, 127, 189–200
- ShadowMapBlur, 408
- ShadowMappingShader, 314
- ShadowMapShader, 409
- SimpleCamera, 288
- SimpleEffect, 147
- SkinnedTangentVertex, 523
- SocketHelper, 457
- Sound, 260–265
- SpaceCamera, 289
- SplashScreen, 410
- SpriteBatch, 35
- StartGameScreen, 476
- StringHelper, 74–77
- TangentVertex, 191
- TestGame, 81
- TestRenderingTrack, 349
- TetrisGame, 93
- TetrisGrid, 94
- TrackImporter, 355
- TrackLine, 349
- UdpTests, 458
- Vector3Helper, 407
- XmlHelper, 356
- XnaShooterGame, 296
- ClearColor, 214**
- ClearDepth, 214**
- CodeRush, 12**
- Collada, 355**
 - Dungeon Quest, 522–526
- ColladaCombiModels class, 356**
- ColladaLoader class, 356**
- ColladaModelFile class, 523**
- ColladaTrack class, 356**
- collision detection**
 - Dungeon Quest, 520
 - physics and, 391–392
- collision testing, Pong, 47–52**
- color correction, 228–231**
- ColorHelper class, 80**
- Components property, 93**
- compression, 247**
- ComputeTangentMatrix method, 186**
- connecting players, 462–464**
- content**
 - accessing, 66–68
 - importing, 66–68
- content directories, 65–66**
 - Rocket Commander, 65
- content pipeline, 8–9**
- Content Processor, 18**
- Content Processor mode, 66**
- coordinate system, 130**
- CreateLookAt method, 129**
- CreateRotation method, 128**
- CreateScale method, 128**

CreateTranslation method

CreateTranslation method, 128

cries of monsters, Dungeon Quest, 531–532

CutExtension method, 75

D

.dds file format, 63

Default Chase Car Game Camera, 378

deploying games, Xbox 360, 141–142

depthBias, 426

detail in objects, 174–178

directional lights, 156

directories

- content directories, 65–66
- Rocket Commander, 65

DirectSound, 244–245

- .wav files, 245–247

DirectSound device, initializing, 244

Displacement Mapping, 174

Draw() method, 7

- Breakout game, 86–87

DrawableGameComponent class, 93

DrawIndexedPrimitives, 168

DrawSprites method, 40, 77–78

Drum Sticks, 270

Dungeon Quest

- abilities, 506–507, 507–508
- AI (Artificial Intelligence), 530
- cameras, 520
- characters, 498–501
- Collada model files, 522–526
- collision detection, 520
- components, 520
- content, 520
- cries of monsters, 531–532
- designing, 493
- editor, 539–543
- effects, 520
- enemies, 501–505

engine

- creating, 518–519
- graphic, 519–522
- fonts, 520
- game handling, 521
- ideas for, 494
- items, 506
- leveling system, 510–513
 - improving, 513–514
- lights, 521, 526–529
- lines, 522
- models, 522
- multiplayer support, 532–533
 - network messages, 533–534
 - Xbox LIVE, 535–538
- path-finding, 530–531
- physics, 522
- scenario, 494–496
- shaders, 522
- sound, 522
- spells, 506–507, 508
 - barbarian, 509
 - hunter, 510
 - mage, 510
 - priest, 509
- story, 496–498
- textures, 522
- used assemblies, 520
- weapons, 506

DXT, 18

DXT compression algorithms, 181

DXT1, 67

dynamic unit tests, 29

E

edge detection shaders, 237–238

EditorScreen class, 539

EffectManager class, 295, 319

EnableDefaultLighting method, 26

EndGame message, 533
EndGameMessage class, 467
engines
 capabilities, 116–117
 physics engines, 372, 379–381
 Ageia PhysX, 381–382
 Bullet Physics Library, 384
 Farseer Physics Engine, 383
 Havok, 382–383
 Newton Game Dynamics, 384
 nV Physics Engine, 384
 ODE (Open Dynamics Engine), 383
 Physics and Math Library, 384
 Tokamak Game Physics, 384
 XPA, 383
 shaders, importing, 164–170
 unit testing, 117–118
entering text, 275–277
EnumHelper class, 79–80
error messages, logging, 68–69
Esc, 17
ExtractFilename method, 75

F
F5, 16
farPlane, 425
Farseer Physics Engine, 383
 .fbx file format, 63
FBX Model Importer, 67
Fight sound, Tetris, 99
file formats, textures, 63
filenames, StringHelper class, 75–76
firewalls, 453–455
floatingGrid, 104
FontToRender class, 97
Free Camera Mode, 378
FX Composer, shaders and, 151–153
.fx files
 layout, 153–55
 parameters, 155–157

G
GAC (Global Assembly Cache), 69
Game class, 6, 92–93
 Draw() method, 7
 Initialize() method, 7
 Update() method, 7
Game namespace, 296
GameAsteroidManager.cs, 296
GameInfoAndPlayerList message, 533
GamePadAJustPressed method, 98
gamePadUp, 47
GameScreens namespace, 296
 Help class, 280–281
GameServer class, 456
GameSync message, 534
GenerateLandscapeHeightFile, 342
GenerateTangents method, 197
GenerateVerticesAndObjects helper method, 360
GetAnimatedTexture method, 304
GetCue method, 261
GetDirectory method, 75
GetFasterAndMoveDown, 323
GetLastChatMessage method, 479
GetOnlineHighscores method, 434
GetTextWidth method, 96
 glass shader, 202
GPU (graphic processing unit), 128
Gran Turismo, 335
Graphics namespace, 296
 graphics, loading, 17
GraphicsDeviceManager, 92
gravitation, 376–379
Guitars, 270

H
 hack and slash role-playing games, 487–491
HandleAsteroidCollision method, 394
HandleKeyboardInput method, 276–277

HandleSectorPhysics method

HandleSectorPhysics method, 393

Havok, 382–383

Help.cs, 296

helper classes, 61

Assert, 71

Canyon Commander, 407

ColorHelper class, 80

EnumHelper class, 79–80

Input class, 97–99

RandomHelper class, 73–74

SpriteHelper class, 77–79

StringHelper class, 74–77

TextureFont class, 95–97

Helpers namespace, 77

Highscore class, 433

Highscores screen, 417–418

HitpointsDecrease message, 533

HLSL (High Level Shader Language), 146

Hooke's law, 390

<http://abi.exdream.com>, 20

<http://creators.xna.com>, 26

http://en.wikipedia.org/wiki/Microsoft_XNA, 20

<http://forums.xna.com>, 26

<http://martinfowler.com>, 33

<http://msdn.microsoft.com/directx/xna>, 20

<http://xnadevelopment.com>, 20

<http://xnaresources.com>, 20

I

IGameScreen, 279–280

importing, shaders to engines, 164–170

In-Game UI, 281–286

Initialize() method, 7

input

Big-Button-Pad, 269

Chatpad device, 269

Drum Sticks, 270

entering text, 275–277

Guitars, 270

Tetris, 107–110

Input class, 97–99, 267, 268–271

mouse rectangles, 272–275

Update method, 271–272

InvalidOperationException, 244

Item class, 325–327

ItemCollected message, 534

J

JoinGame message type, 533

JoinGameMessage class, 458, 464

JoinSession method, 537

.jpg file format, 63

K

KaktusSeg.dds, 175

KaktusSegNormal.dds, 175

KeyboardSpaceJustPressed property, 98

KeyboardState struct, 275

keyboardStateLastFrame variable, 275

L

landscape, 333, 338–339

rendering, 340–345

textures, 339–340

unit testing, 342

Landscape class, 342

LandscapeGridHeights.png, 339

LandscapeNormalMapping shader, 344

Landscapes namespace, 340

laws of motion, 370–371

left hand coordinate system, 130

LensFlare.cs, 296, 407

Level.cs, 296

LevelGroundRender class, 540

LevelUp message, 534

lighting

- directional lights, 156
- Dungeon Quest, 526–529
- point lights, 156
- spot lights, 156

lightVec variable, 186

LineKill sound, Tetris, 99

LineManager3D class, 125

LineRendering.fx shader, 126, 127

lines, rendering in 3D models, 123–127

links, bookmarks, 21–22

lists, StringHelper class, 76–77

Load method, 352–353

LoadAnimation method, 525

LoadBones method, 525

LoadGraphicsContent method, 7

loading, textures, 36

LoadLevel class, 343

LoadLevel method, 343

LoadMeshes method, 525

Log class, 68

Log.txt file, 68

Log4Net, 68

loopings, 385–387

Lose sound, Tetris, 99

M

MainMenu Run method, 299

MakeScreenshot method, 139

mapping

- Displacement Mapping, 174
- Normal Mapping, 174
- Offset Mapping, 174
- Parallax Mapping, 174

marble.dds texture, 163

Material class, 190

MDX (Managed DirectX), 24

menu effects, 231–232

menu sounds, 265

MeshHelper class, 197

MeshRenderManager, 416

messages, network messages, 464–470

methods

- AddChatMessage, 479
- AddEffect, 319
- AddSpriteToRender, 122
- ApplyGravity, 378
- ApplyGravityAndCheckForCollisions, 395
- BeginReceiveFrom, 460
- BeginSendTo, 461
- CalcLandscapePos, 344
- CalcVectors, 307
- CheckBallCollisions, 86
- ComputeTangentMatrix, 186
- CreateLookAt, 129
- CreateRotation, 128
- CreateScale, 128
- CreateTranslation, 128
- CutExtension, 75
- Draw(), 7
- DrawSprites, 40, 77–78
- EnableDefaultLighting, 26
- ExtractFilename, 75
- GamePadAJustPressed, 98
- GenerateTangents, 197
- GenerateVerticesAndObjects, 360
- GetAnimatedTexture, 304
- GetCue, 261
- GetDirectory, 75
- GetLastChatMessage, 479
- GetOnlineHighscores, 434
- GetTextWidth, 96
- HandleAsteroidCollision, 394
- HandleKeyboardInput, 276–277
- HandleSectorPhysics, 393
- Initialize(), 7
- JoinSession, 537
- KeyboardSpaceJustPressed, 98

methods (*continued*)

methods (*continued*)

Load method, 352
 LoadAnimation, 525
 LoadBones, 525
 LoadGraphicsContent, 7
 LoadLevel, 343
 LoadMeshes, 525
 MakeScreenshot, 139
 OnHandleMessage, 479
 OnReceive, 479
 Play, 260
 ReceiveDataFrom, 461
 Render, 125, 132–134
 RenderBall(), 42–43
 RenderBillboards, 307
 RenderCar, 412
 RenderCentered, 87
 RenderMenuBackground, 412
 RenderModel, 166
 RenderOnScreen, 118
 RenderPaddles(), 43
 RenderSky, 212
 Resolve, 220
 ResolveBackBuffer, 139
 Select, 304
 SendChatMessage, 481
 SetParameters, 190
 SetRenderTarget, 220
 Show, 223
 ShowGroundGrid, 350
 ShowLives, 42
 ShowTrackLines, 351
 SpriteBatch, 36
 StartLevel, 84
 StartMusic, 255
 StartNewBall, 47–48
 StartTest, 38
 TestBackgroundBoxes, 100
 TestBallCollisions, 49–50
 TestGameSprites(), 41–42

TestGetAllEnumNames, 79–80
 TestMenuSprites, 37, 38
 TestNormalMappingShader, 193–194
 TestRenderModel, 135
 TestSimpleShader, 165
 TestSkyCubeMapping, 211–212
 TestSounds(), 54
 TestUdpClient, 458
 TestUdpServer, 458
 tex2D, 163
 TryParse, 56
 Update(), 7
 UpdateVertexBuffer, 125
 Write(), 69
 WriteArrayData, 80
 WriteText, 96
 XnaTexture, 220

Mission class, 312

MissionSelection.cs, 296

Model class, 132

motion blur, 227–228

mouse, Xbox 360, 268

mouse rectangles, 272–275

moveFactorPerSecond, 20, 50

MoveTypes enum, 107–108

multiplayer application, writing, 483

Multiplayer class, 457, 470

multiplayer games, role-playing, 491–493

multiplayer support, Dungeon Quest, 532–533

network messages, 533–534

Xbox LIVE, 535–538

music, file format, 63

N

NATs (Network Address Translators), 453–455

nearPlane, 425

Need for Speed, 335–336

.NET 2.0 Framework, 12**networks**

- architecture, 455–458
- chat application
 - chat messages, 479–482
 - final application, 482–483
 - game screens, 474–479
- connecting two players, 462–464
- messages, 464–470
 - multiplayer support, Dungeon Quest, 533–534
- server game list, 470–473

Newton Game Dynamics, 384**Newton's laws of motion, 370–371****NextBlock class, 105****NextBlock game component, 104, 111****Normal Mapping**

- pixels, 176
- shaders, 174
- textures, 175
- unit testing, ShaderEffect Class, 193–195

NormalMapCompressor, 180**normalVector, 187****NumbersFont class, 282****NumbersFont.png texture, 282****NumOfBlockTypes, 104****NUnit Framework, 70–71****NUnit GUI, 71–72****nV Physics Engine, 384****O****objects, detail, 174–178****ODE (Open Dynamics Engine), 383****Offset Mapping, 174, 201–202****OnHandleMessage method, 479****OnReceive method, 479****Options.cs, 296****oscilloscope, Pong and, 29****P****Parallax Mapping, 174, 201–202, 204****ParallaxMapping class, 407****ParallaxMapping.fx shader, 119, 180****ParallaxShader.cs, 296****parameters**

- .fx files, 155–157
- shaders, 166–167

path-finding, Dungeon Quest, 530–531**physics, 369–370**

- car collision, 395–398
- car physics, 375
- collision detection, 391–392
- Dungeon Quest, 522
- gravitation, 376–379
- loopings, 385–387
- spring physics, 388–391

Physics and Math Library, 384**physics engines, 372, 379–381**

- Ageia PhysX, 381–382
- Bullet Physics Library, 384
- Farseer Physics Engine, 383
- Havok, 382–383
- Newton Game Dynamics, 384
- nV Physics Engine, 384
- ODE (Open Dynamics Engine), 383
- Physics and Math Library, 384
- Tokamak Game Physics, 384
- XPA, 383

PhysicsAsteroidmanager class, 392–395**PhysicsAsteroidManager.cs, 296****Pixel Shader, 173****pixel shader, 162–164, 187–189****pixels, Normal Mapping, 176****planning, 30–31**

- initial difficulties, 33–34
- Pong, 31–32

Play method, 260**PlayCue, 261**

player class

player class, 457
player input, Pong, 44–47
PlayerDied message, 533
players, connecting, 462–464
PlayerUpdate message, 533
.png file format, 63
point lights, 156
Pong
 ball, adding, 41–44
 collision testing, 47–52
 improving, 57
 original, 29
 paddles, adding, 41–44
 planning, 31–32
 initial difficulties, 33–34
 player input, 44–47
 sound, 53–55
 sprites, 35–36
 TestSingleplayerGame, 47
 textures, 34
 troubleshooting, 58–59
 unit testing, 37–74
PongBallHit.wav, 35, 53
PongBallLost.wav, 35, 53
post-screen shaders
 implementation
 PostScreenDarkenBorder class, 221–224
 RenderToTexture class, 220–221
 unit testing, 224–225
 sky cube mapping, 208–213
 writing, 214–217
PostScreenDarkenBorder class, 221–224
PostScreenDarkenBorder.fx, 214
PostScreenGlow, 225
PostScreenGlow class, 407
postScreenShader variable, 222
PostScreenShadowBlur.fx shader, 424
PreScreenSkyCubeMapping.cs, 296, 407
PreScreenSkyCubeMapping.fx, 210
Principia Mathematica, 371

Projectile class, 323–325
ProjectionMatrix, 129
projects, new, 15
properties, Components, 93

Q

quitting program, 17

R

racing simulations, 334–335
 Gran Turismo, 335
 Need for Speed, 335–336
 Trackmania, 336–337
RandomHelper class, 73–74
ReceiveDataFrom method, 461
reflection, 203
Reflector, 6
RelativeScreenRect class, 478
remDownPressed, 47
Render method, 125, 132–134
RenderBall() method, 42–43
RenderBillboards method, 307
RenderCar method, 412
RenderCentered method, 87
rendering
 lines, 3D models, 123–127
 shaders and, 167–168
 textures, 3D models, 121–123
RenderLandscapeBackground, 313
RenderMenuBackground method, 412
RenderModel method, 166
RenderOnScreen method, 118
RenderPaddles() method, 43
RenderSky method, 212
RenderTarget class, 219
RenderToTexture class, 220–221
Resolve method, 220
ResolveBackBuffer method, 139

ResourceName annotation, 157

Rocket Commander

- directories, 65
- MDX, 234–235
- PhysicsAsteroidManager class, 392–395
- Rocket Motor sound, 263
- screenshots, 235–237
- threads, 235
- XNA and, 232–234

Rocket.dds, 119

RocketCommanderGame class, 296

RocketCommanderXna namespace, 297

RocketHeight.dds, 119

RocketNormal.dds, 119

role-playing games, 485

- types, 486–487
 - hack and slash, 487–491
 - multiplayer, 491–493

S

ScreenBorderFadeout.dds, 215

ScreenHeight property, 478

ScreenPositionX property, 478

ScreenPositionY property, 478

screens, 277

- background, 278
- Help, 280–281
- In-Game UI, 281–286

ScreenshotCapturer class, 139

ScreenWidth property, 478

scrolling, background, 20

scrollPosition, 20

Select method, 304

SendChatMessage method, 481

server, game list, 470–473

ServerGame class, 457

SessionConnection class, 456, 462

SetParameters method, 190

SetRenderTarget method, 220

Shader Model, 173

ShaderEffect class, 127, 189–191

- Normal Mapping unit test, 193–195
- tangent data, custom processor and, 195–199
- TangentVertex class, 191–193

shaders, 145. See also Pixel Shader

- Canyon Commander, 407
 - compiling, 165–166
 - Dungeon Quest, 522
 - edge detection, 237–238
 - example games, 148–149
 - .fx files, layout, 153–55
 - glass shader, 202
 - history of, 146–147
 - importing to engine, 164–170
 - LandscapeNormalMapping, 344
 - LineRendering.fx, 126, 127
 - Normal Mapping shaders, 174
 - offset mapping, 201–202
 - overview, 182–185
 - parallax mapping, 201–202
 - ParallaxMapping.fx, 119
 - parameters, 166–167
 - pixel shader, 162–164, 187–189
 - post-screen shaders
 - implementation, 220–225
 - sky cube mapping, 208–213
 - writing, 214–217
 - reflection, 203
 - rendering with, 167–168
 - testing, 169–170
 - vertex formats, 167
 - vertex input, format, 158–159
 - vertex shader, 159–161, 185–186
 - matrices, 185–186
 - water, 203
- Shaders namespace, 296**
- shadowColor, 426**
- ShadowMapBlur class, 408**

shadowMapDepthBias

shadowMapDepthBias, 426
ShadowMappingShader class, 314
ShadowMapShader class, 409
shipRotation, calculating, 318
Show method, 223
ShowGroundGrid method, 350
ShowLives method, 42
ShowTrackLines method, 351
SimpleCamera class, 288
SimpleEffect class, 147
SimpleShader.fx, 150
 file layout, 153–55
 FX Composer, 151–153
SizeType parameter, 220
SkinnedTangentVertex class, 523
skyCube variable, 222
SmallAsteroid.cs, 296
SocketHelper class, 457
sound
 compression, XACT, 252–254
 Dungeon Quest, 522
 file format, 63
 menu sounds, 265
 Pong, 53–55
 XACT and, 243
 cues, 256–257
 XnaShooter, 297–298
Sound Bank, 54
Sound class, **Play** method, 260
Sound Cues, 54
space camera, 289–292
SpaceBackground.dds texture, 36
SpaceCamera class, 289
SpaceCamera.cs, 296
SpeedOfSound variable, 249
SplashScreen class, 410
spot lights, 156
spring physics, 388–391
SpriteBatch class, 35
SpriteBatch methods, 36
SpriteHelper class, 77–79

sprites, Pong, 35–36
StartGame message, 533
StartGameScreen class, 476
StartLevel method, 84
StartMusic method, 255
StartNewBall method, 47–48
StartTest method, 38
static unit tests, 29
StreamWriter, 68
StringHelper class, 74–77

T

tangent data, **ShaderEffect** class,
 195–199
TangentVertex class, 191
TCP, **UDP** and, 450–453
TestAsteroidPhysicsSmallScene unit
 test, 392
TestBackgroundBoxes method, 100
TestBallCollisions, 49–50, 83
TestCarPhysicsOnPlane unit test, 376
TestCarPhysicsOnPlaneWithGuardRails
 unit test, 375
TestCreateAndJoinGame unit test, 472
TestDriven.NET, 12, 24, 70–71
TestFallingBlockAndLineKill unit test,
 105, 110
TestGame class, 81
TestGameSprites method, 41–42
TestGameSprites test, 83
TestGameSprites() method, 41
TestGetAllEnumNames() method, 79–80
TestGetListOfGames unit test, 471
testing
 shaders, 169–170
 unit testing, 37–74
TestMenuSprites method, 37, 38
testModel variable, 222
TestNormalMappingShader method,
 193–194
TestPlaySounds unit test, 298

TestRenderAnimatedModel unit test, 524
TestRenderingTrack class, 349
TestRenderLandscape unit test, 342
TestRenderLandscapeBackground unit test, 313
TestRenderLines unit test, 124
TestRenderModel method, 135
TestRotatingBlock unit test, 110
TestShowBones unit test, 524
TestSimpleShader method, 165
TestSingleplayerGame, 47
TestSingleplayerGame unit test, 55
TestSkyCubeMapping method, 211–212
TestSounds, 83
TestSounds() method, 54
TestUdpClient method, 458
TestUdpServer method, 458
Tetris
 background, rendering, 100–102
 block types, 102–105
 BlockFallDown sound, 99
 BlockMove sound, 99
 BlockRotate sound, 99
 blocks, gravity, 105–107
 Fight sound, 99
 input handling, 107–110
 LineKill sound, 99
 Lose sound, 99
 NextBlock game component, 111
 TestFallingBlockAndLineKill unit test, 105
 Victory sound, 99
TetrisGame class, 93
TetrisGrid class, 94
tex2D method, 163
texelHeight, 425
texelWidth, 425
texOffsetX, 425
texOffsetY, 425
text, entering, 275–277
TextureFont class, 95–97, 96
 implementation, 96–97

textures
 background, displaying, 19
 Dungeon Quest, 522
 file formats, 63
 landscapes, 339–340
 loading, 18, 36
 imported, 18
 Normal Mapping, 175
 Pong, 34
 rendering, 3D models, 121–123
 SpaceBackground.dds, 36
 XnaShooter, 300–301
 animated, 303–304
.tga file format, 63
threads, Rocket Commander and, 235
Tokamak Game Physics, 384
track, vertices, 358–363
TrackImporter class, 355
TrackLine class, 349
Trackmania, 336–337
tracks
 importing data, 356–358
 interpolating splines, 352–354
 unit testing, 349–352
Tracks namespace, 340
TransformPosition, 184
troubleshooting, 27–28
 Pong, 58–59
TryParse method, 56

U

UDP
 sending data, 458–461
 TCP and, 450–453
UdpTests class, 458
unit testing, 37–74, 69–70
 3D models, 131–135
 Breakout, 82–83
 Canyon Commander, 408, 418–420
 engines, 117–118

unit testing *(continued)*

unit testing *(continued)*

- landscape, 342
- NUnit Framework, 70–71
- post-screen shaders, 224–225
- rules for, 73
- starting unit tests, 71–72
- TestDriven.Net, 70–71
- TestRenderLines, 124
- tracks, 349–352

UnitDied message, 533, 534

Update() method, 7

- Input class, 271–272

UpdateVertexBuffer method, 125

usePointLights variable, 529

user interface

- In-Game UI, 281–286
- XnaShooter, 298–300

V

VBScreenHelper, 212

Vector3Helper class, 407

vertex formats, 167

vertex input format, 158–159

vertex shaders, 159–161

- matrices, 185–186
- problems with, 177

VertexBuffers, 213

VertexElement method, parameters, 193

VertexInput structure, 160

VertexOutput structure, 158

VertexOutput_Specular20 structure, 187

VertexPositionColor, 125

VertexPositionNormalTexture, 158, 167

vertices, track, 358–363

Victory sound, Tetris, 99

viewVec variable, 186

virtualLightDistance, 424

virtualVisibleRange, 424

Visual Studio 2005 Professional, 12–13

W

water, 203

.wav files, 53, 245–247

Wave Bank, 54

websites

- <http://abi.exdream.com>, 20
- <http://creators.xna.com>, 26
- http://en.wikipedia.org/wiki/Microsoft_XNA, 20
- <http://forums.xna.com>, 26
- <http://martinfowler.com>, 33
- <http://msdn.microsoft.com/directx/xna>, 20
- <http://xnadevelopment.com>, 20
- <http://xnaresources.com>, 20
- www.AIWisdom.com, 530
- www.DungeonQuestGame.com, 517
- www.riemers.net, 20
- www.thezbuffer.com, 20
- www.ziggyware.com, 20

WorldMatrix, 128

worldToTangentSpace matrix, 186

worldViewProj matrix, 155

Write() method, 69

WriteArrayData method, 76, 80

WriteOnly, VertexBuffers and, 213

WriteText method, 96

www.AIWisdom.com, 530

www.DungeonQuestGame.com, 517

www.riemers.net, 20

www.thezbuffer.com, 20

www.ziggyware.com, 20

X

.x model files, 63

X Model Importer, 67

X3Audio, 264

X3AudioListener, 264

X3DAudio, 264

X3DAudioListener, 264

XACT, 53

- compression, 252–254
- project creation, 249–250
- sound and, 243
 - cues, 256–257
- Sound Bank, 54
- sound banks, 254–256
- Sound Cues, 54
- Wave Bank, 54
- wave bank, creation, 250–251
- XACT Audio Project, .xap files, 63
- XNA and, 248
 - changes in XNA 2.0, 259–260

.xap file format, 63, 248**XAudio, 259****Xbox 360**

- configuration, 13–15
- console debugging, 142
- debugging, 56
- deploying games, 141–142
- dlls, external, 56
- Dungeon Quest, 535–538
- game component, writing, 143
- gamepads, 56
- loading content directly, 56
- mouse functionality, 268
- save games, 56
- test resolutions, 56
- testing, 56
- unit testing, 56
- Xbox Live Service, 13
- XNA Game Launcher, downloading, 140–141

XmlHelper class, 356**XNA 2.0, 25–27**

- networking, 449–450
- new features, 448–449
- TCP and UDP, 450–453

XNA Content Pipeline, 5, 8–9, 23–24

- advantages, 64–65
- disadvantages, 64–65
- management, 62–63

XNA Framework, 3–4

- application model, 5–8
- DirectX and, 3–4
- Xbox 360, 13–15

XNA Game Application Model, 5**XNA Game Launcher, downloading, 140–141****XNA Game Studio Express, 4–5**

- installation, 10–12
- requirements, 10

XNA Graphic Engine, 5**XNA Racing Game, 333, 337–338.**

See also Canyon Commander

XNA Shooter, 295

- 3D effects, 319–320
- asteroids, 303
- Corvette, 301
- Firebird, 303
- game logic, 317–319
- Item class, 325–327
- OwnShip, 301
- Projectile class, 323–325
- Rocket-Frigate, 303
- Small-Transporter, 302
- unit class, 320–323

XnaGraphicEngine, 155**XnaShooter, 296**

- 3D models, 301–303
- billboards, 305–308
- landscape rendering, 308–315
- sounds, 297–298
- textures, 300–301
 - animated, 303–304
- user interface, 298–300

XnaShooter namespace, 297**XnaShooter.xap, 297****XnaShooterGame class, 296****XnaShooterGameMusic, 254****XnaTexture method, 220**

- XPA physics engine, 383



