

Inhaltsverzeichnis

V	Vorwort	17
V.1	An wen richtet sich dieses Buch?	18
V.2	Ansprüche an den Computer	18
V.3	Website	19
V.4	Bildergalerie	19
V.5	Über den Autor	20
1	Einführung	21
1.1	Installation	21
1.2	Projekttypen	22
1.3	XNA Creators Club	23
1.4	Das XNA Framework im Überblick	24
1.4.1	Platform	25
1.4.2	Core Framework	25
1.4.3	Extended Framework	26
1.4.4	Games	27
1.5	Das Fundament eines Spiels: Die Game-Klasse	27
1.5.1	Das ereignisbasierte Modell	28
1.5.2	Der Game Loop	28
1.5.3	Die gameTime-Struktur	29
1.5.4	Im Fokus: Die Draw()-Methode	30
1.6	Fullscreen	31
1.6.1	Das Dilemma mit den Auflösungen	31
1.6.2	Save Regions	32
1.7	Zusammenfassung	33

2	Zweidimensionale Grafikausgabe	35
2.1	Grundlagen	35
2.1.1	Der Ursprung eines Sprites	36
2.1.2	Sprites rendern	37
2.1.3	Transparenz	42
2.2	Game Components	43
2.2.1	Nichtgrafische Komponenten	44
2.2.2	Die Game.Components-Auflistung	44
2.2.3	Game Services	45
2.3	Transformationen	47
2.3.1	Rotation	47
2.3.2	Vor- und Rückbewegung	49
2.3.3	Die Komponente im Überblick	49
2.4	Sprite-Animationen	51
2.4.1	Der Konstruktor	51
2.4.2	Grafische Ressourcen laden	52
2.4.3	Selektion des nächsten Animationsframes	53
2.4.4	Animation rendern	54
2.5	Background Scrolling	55
2.6	Text ausgeben	57
2.6.1	Eine FpsCounter-Komponente	61
2.7	Zusammenfassung	63
3	Interaktionen	65
3.1	XINPUT	65
3.2	Xbox 360 Controller	65
3.2.1	Force Feedback	68
3.2.2	Die VibrationManager-Komponente	68
3.3	Tastaturstatus ermitteln	72
3.4	Ansprechen der Maus	72
3.5	Lenkräder, Gitarren & Co	73
3.6	Xbox Guide Interface	73
3.6.1	Eingabeaufforderung	74
3.6.2	Message Boxen	75
3.6.3	Storage Device Selector	77

3.7	Storage Devices	77
3.7.1	Title Storage	77
3.7.2	User Storage	77
3.7.3	Dateien auflisten	78
3.7.4	Dateien lesen und ändern	78
3.7.5	Daten (de)serialisieren	79
3.8	Game States	80
3.8.1	Anforderungsanalyse	80
3.8.2	Die Basisklasse: Screen	81
3.8.3	Das Oberhaupt: Die ScreenManager-Klasse	84
3.8.4	Hilfsklasse zur Verarbeitung von Benutzereingaben	87
3.8.5	Implementierung eines Cursors	89
3.8.6	Exemplarische Implementierung	90
3.9	Kollisionskontrolle	92
3.9.1	Bounding Boxes	93
3.9.2	Kollisionskontrolle auf Pixelbasis	94
3.9.3	Per-Pixel-Kollisionskontrolle bei transformierten Objekten	97
3.10	2D-Partikelsysteme	103
3.10.1	Eigenschaften eines Partikels	103
3.10.2	Fundament: Die ParticleSystem-Klasse	105
3.10.3	Beispiel: Explosion	110
3.10.4	Beispiel: Fahrzeugspuren	112
3.11	Das erste Spiel: 2D Airplane Shooter	115
3.11.1	Übersicht der Screens	116
3.11.2	Darstellung der Gegner	117
3.12	Zusammenfassung	118
4	3D-Grundlagen	119
4.1	Vertices und Primitive	119
4.2	Koordinatensysteme	119
4.2.1	Objektkoordinaten (Object Space)	120
4.2.2	Weltkoordinaten (World Space)	121
4.2.3	Sichtkoordinaten (Camera Space)	121
4.2.4	Bildschirmkoordinaten (Screen Space)	121
4.3	Der Vertex Buffer	121
4.3.1	Initialisierung des Vertex Buffers	122
4.3.2	Vertex Buffer mit Daten bestücken	123
4.3.3	Vertex Declaration	123

4.4	Kameraeinstellung	125
4.4.1	View Matrix	126
4.4.2	Projection Matrix	126
4.4.3	World Matrix	127
4.5	Rendering mithilfe der BasicEffect-Klasse	128
4.5.1	Initialisierung und Konfiguration	129
4.5.2	Primitive rendern	129
4.6	Render States	131
4.6.1	Back Face Culling	131
4.6.2	Fill Mode	132
4.7	Der Tiefenspeicher (Z-Buffer)	132
4.7.1	Z-Buffer (de)aktivieren	133
4.7.2	Z-Buffer „säubern“	134
4.8	Der Index Buffer	135
4.8.1	Initialisierung	136
4.8.2	Indizes definieren	136
4.8.3	Geometrie mit Index Buffern rendern	137
4.8.4	Indexed Primitives in Szene gesetzt	139
4.9	Zusammenfassung	141
5	Texturen, Beleuchtung und 3D-Modelle	143
5.1	Texturen	143
5.1.1	Texturkoordinaten	144
5.1.2	Texturen laden und anwenden	146
5.1.3	Texturadressierung	147
5.1.4	Mipmaps	151
5.1.5	Pixelformate	152
5.1.6	Texturfilter	152
5.1.7	Das additive Farbmodell	157
5.1.8	Alpha Blending	158
5.1.9	Alpha Testing	164
5.1.10	Ein Surface als Render-Ziel verwenden (Render to Texture)	165
5.1.11	Erstellen von Screenshots	171
5.2	Materialien und Beleuchtung	172
5.2.1	Materialeigenschaften	174
5.2.2	Direktionale Lichtquellen	175

5.3	Die virtuelle Kamera	175
5.3.1	Das Fundament einer virtuellen Kamera	176
5.3.2	First-Person-Kamera	176
5.4	Billboards	179
5.4.1	Theorie	180
5.4.2	Implementierung	181
5.5	Meshes	183
5.5.1	Aufbau eines 3D-Modells	184
5.5.2	Modelle laden	185
5.5.3	Modell rendern	186
5.6	Nebel	189
5.6.1	Linearer Nebel	190
5.6.2	Exponentieller Nebel	191
5.6.3	Vertex Fog versus Pixel Fog	191
5.6.4	Range-Based Fog	191
5.7	Antialiasing (Multisampling)	192
5.7.1	Multisampling aktivieren	193
5.8	Zusammenfassung	195
6	Die Content Pipeline	197
6.1	Architektur der Content Pipeline	197
6.2	Content Processor zur Generierung eines Terrains	199
6.2.1	Konvertierung der Textur	201
6.2.2	Aufspannen des Drahtgittermodells	202
6.2.3	Materialeigenschaften	202
6.2.4	Vertex Channels	203
6.2.5	Primitive festlegen	204
6.2.6	Konvertierung der Modelldaten	205
6.2.7	Content Processor verwenden	205
6.3	Content-Pipeline-Komponenten debuggen	206
6.4	Parametrisierter Content Processor	207
6.4.1	Einem Content Processor Parameter übergeben	209
6.5	Sky Box	209
6.5.1	XML Content Importer verwenden	210
6.5.2	Content Processor: Sky Box	212
6.5.3	Content Compiler: Sky Box	213

6.5.4	Content Reader: Sky Box	215
6.5.5	Sky Box rendern	217
6.6	Linsenreflektionen (Lens Flares)	218
6.6.1	Elemente	219
6.6.2	Theorie	219
6.6.3	Der Content Importer	223
6.6.4	Der Content Processor	226
6.6.5	Die Content Compiler	226
6.6.6	Der Content Reader	227
6.6.7	Lens-Flare-Effekt rendern	228
6.6.8	Occlusion Queries	230
6.7	Quadtree	235
6.7.1	Der Content Importer	236
6.7.2	Der Content Processor	238
6.7.3	Content Compiler	248
6.7.4	Content Reader	250
6.7.5	Die Quadtree-Klasse	253
6.7.6	Brute Force Rendering	254
6.7.7	View Frustum Culling	254
6.7.8	Kollisionskontrolle mit dem Terrain	256
6.8	Picking	258
6.8.1	Grundlagen	259
6.8.2	Picking auf 3D-Modelle ausführen	260
6.8.3	Picking auf Primitive-Ebene	261
6.9	Custom Model Processor	270
6.9.1	Die CustomModelContent-Klasse	271
6.9.2	Custom Model Content Processor	272
6.9.3	Der Content Compiler	274
6.9.4	Der Content Reader und die CustomModel-Klasse	275
6.10	Automatische Serialisierung	278
6.11	Zusammenfassung	280
7	Stencil Buffer	281
7.1	Initialisierung des Stencil Buffers	282
7.2	Der Stencil-Test	283
7.2.1	Reference Stencil	284
7.2.2	Stencil Mask und Stencil Write Mask	284

7.2.3	Vergleichsoperatoren	284
7.2.4	Aktualisierung des Stencil-Buffer-Inhalts	285
7.2.5	Der Stencil Buffer in Aktion	286
7.3	Realisierung eines Spiegels	288
7.3.1	Überblick	288
7.3.2	Back Buffer und Stencil Buffer beschreiben	289
7.3.3	Reflektion rendern	290
7.4	Zusammenfassung	292
8	Shader-Programmierung	293
8.1	High Level Shader Language	294
8.1.1	Vertex Shader	294
8.1.2	Pixel Shader	295
8.1.3	Shader Models	295
8.1.4	Schlüsselwörter	296
8.1.5	Primitive Datentypen	296
8.1.6	Vektoren und Matrizen	297
8.1.7	Swizzling	298
8.1.8	Schreibmasken	298
8.1.9	Strukturen	299
8.1.10	Typkonvertierung	299
8.1.11	Modifizierer	301
8.1.12	Funktionen	302
8.1.13	Programmablaufsteuerung	303
8.1.14	Integrierte Funktionen	304
8.2	Effekte im XNA Framework	305
8.2.1	Der erste Vertex Shader	306
8.2.2	Der erste Pixel Shader	308
8.2.3	Renderoptionen (Technique)	310
8.2.4	Die Effect-Klasse	312
8.2.5	Effect Pools	314
8.3	Lichtquellen	318
8.3.1	Grundbeleuchtung (Ambient Lighting)	319
8.3.2	Direktionale Lichtquellen	319
8.3.3	Specular Highlights	323
8.3.4	Punktlichter (Point Lights)	327
8.3.5	Spotlichter	331

8.4	Bump Mapping	334
8.4.1	Content Processor zur Aufbereitung der Daten	336
8.4.2	Der Content Compiler	340
8.4.3	Globale Variablen der Effekt-Datei	340
8.4.4	Der Vertex Shader	341
8.4.5	Der Pixel Shader	342
8.5	Sky Boxes mit Cube Maps	343
8.5.1	Der Content Processor	344
8.5.2	Die SkyBox-Klasse	346
8.5.3	Globale Variablen der Effekt-Datei	347
8.5.4	Der Vertex Shader	347
8.5.5	Der Pixel Shader	348
8.6	Environment Mapping	348
8.6.1	Dynamische Cube Map erzeugen	349
8.6.2	Globale Variablen der Effekt-Datei	351
8.6.3	Der Vertex Shader	351
8.6.4	Der Pixel Shader	352
8.7	Texture Splatting	352
8.8	Gräser	355
8.8.1	Der Content Processor	356
8.8.2	Globale Variablen der Effekt-Datei	360
8.8.3	Der Vertex Shader	361
8.8.4	Der Pixel Shader	362
8.9	Wasser	362
8.9.1	Reflektionen	363
8.9.2	Clip Planes	365
8.9.3	Projektion der Reflektion	366
8.9.4	Simulation der Wellenbewegung	367
8.9.5	Lichtbrechung (Refraction)	368
8.9.6	Der Render-Vorgang aus Anwendungssicht	370
8.9.7	Fresnelsche Formel	372
8.9.8	Zusammenfassung	374
8.10	Partikelsysteme (Point Sprites)	378
8.10.1	Die ParticleSystem-Klasse im Überblick	379
8.10.2	Dynamische Vertex Buffer	380
8.10.3	Neue Partikel hinzufügen	381
8.10.4	Partikelsystem aktualisieren	383
8.10.5	Den Partikeleffekt rendern	385

8.10.6	Point Sprites	388
8.10.7	Der Vertex Shader	389
8.10.8	Der Pixel Shader	392
8.10.9	Anwendungsbeispiel: Rauch	393
8.10.10	Anwendungsbeispiel: Explosion	395
8.11	Instancing	397
8.11.1	Hardware Instancing	398
8.11.2	Shader Instancing	403
8.11.3	VFetch Instancing	405
8.12	Shadow Mapping	407
8.12.1	Kameraperspektive transformieren	408
8.12.2	Shadow Map rendern	410
8.12.3	Szene schattieren	414
8.13	Post Processing Effects	416
8.14	Post Processing: Kamerawackeln	418
8.14.1	Globale Variablen der Effekt-Datei	418
8.14.2	Der Pixel Shader	419
8.15	Post Processing: Schwarz-Weiß-Szene	419
8.16	Post Processing: 2D-Druckwelle (Distortion)	420
8.16.1	Globale Variablen der Effekt-Datei	421
8.16.2	Der Pixel Shader	422
8.17	Post Processing: Bloom	423
8.17.1	Die Bloom-Klasse	424
8.17.2	Helle Anteile extrahieren	425
8.17.3	Gaussian-Blur-Filter	426
8.17.4	Szenenkomposition	430
8.18	Post Processing: Light Shafts	432
8.18.1	Grundlagen	433
8.18.2	Multiple Render Targets (MRT)	434
8.18.3	Globale Variablen der Effekt-Datei	437
8.18.4	Der Pixel Shader	438
8.19	Zusammenfassung	440

9	Medienwiedergabe	441
9.1	XACT	441
9.1.1	Zweidimensionale Audiowiedergabe	442
9.1.2	In Memory versus Streaming	447
9.1.3	Kategorien	449
9.1.4	Effekte	450
9.1.5	Kompression	450
9.1.6	Dreidimensionale Audiowiedergabe	451
9.1.7	Variablen	453
9.1.8	Runtime Parameter Controls (RPC)	455
9.2	SoundEffect-API	457
9.3	Media Library	460
9.3.1	Alben, Playlists und Songs listen	460
9.3.2	Wiedergabe von Songs aus der Medienbibliothek	461
9.3.3	Songs importieren	462
9.4	Videos	462
9.5	Zusammenfassung	465
10	Multiplayer	467
10.1	Netzwerke	467
10.2	Architekturen (Netzwerktopologien)	468
10.3	Client/Server	468
10.4	Peer-to-Peer	469
10.5	Sessions	470
10.5.1	Session anlegen	472
10.5.2	Sessions suchen und beitreten	474
10.5.3	Paketverluste und Antwortzeiten	475
10.5.4	Session-Ereignisse	476
10.5.5	Nachrichten versenden	477
10.5.6	Nachrichten empfangen	478
10.5.7	Lobbies	478
10.5.8	Sprachübertragung (Voice Support)	479

10.6	Gamer Services	479
10.6.1	Spieleranmeldung	480
10.6.2	Avatare	480
10.6.3	Spielerprofile	480
10.6.4	Anwesenheitsinformationen (Presence Information)	483
10.6.5	Einladungen (Game Invitations)	484
10.6.6	LIVE Party Support	485
10.7	Zusammenfassung	486
A	Indie Games	487
A.1	XNA Creators Club	487
A.2	Verbindung mit der Xbox 360 aufbauen	488
A.3	Debugging	491
A.4	Deployment	492
A.4.1	XNA Creators Club Game Package	492
A.4.2	ClickOnce	492
A.5	Computerspiel veröffentlichen und Geld verdienen	494
A.5.1	Der Veröffentlichungsprozess	494
A.5.2	Bezahlung	495
A.5.3	Demomodus	495
	Stichwortverzeichnis	497