

# Contents

<b>Preface</b>	<b>xiii</b>
<b>About the Authors</b>	<b>xvii</b>
<b>Introduction</b>	<b>xxi</b>
<b>Introduction to the DirectX High Level Shading Language</b>	<b>1</b>
Craig Peeper and Jason L. Mitchell	
Introduction . . . . .	1
A Simple Example . . . . .	2
Assembly Language and Compile Targets. . . . .	4
Hardware Realities . . . . .	6
Compilation Failure . . . . .	6
The Command-line Compiler — fxc . . . . .	7
Language Basics . . . . .	8
Keywords . . . . .	8
Data Types . . . . .	9
Type Modifiers . . . . .	12
Storage Class Modifiers . . . . .	13
Initializers . . . . .	14
Working with Vectors. . . . .	14
Constructors . . . . .	15
Type Casting . . . . .	15
Structures . . . . .	17
Samplers . . . . .	17
Intrinsics . . . . .	19
Math Intrinsics . . . . .	20
Texture Sampling Intrinsics . . . . .	23
Shader Inputs. . . . .	25
Uniform Input . . . . .	25
Varying Input . . . . .	27
Shader Outputs. . . . .	29
An Example Shader. . . . .	31
Optimization . . . . .	39
Matrix Data Type Usage . . . . .	40

Integer Data Type Usage . . . . .	41
Flow Control and Performance . . . . .	42
Importance of Input Type Declarations . . . . .	44
Precision Issues (logp, exp, lit) . . . . .	45
Using the ps_1_x Compile Targets . . . . .	46
Strategy for Targeting ps_1_x . . . . .	51
Integration into an Engine Using D3DX Effects . . . . .	51
Effect Files . . . . .	52
The Effect API . . . . .	57
Integration into an Engine without Using D3DX Effects . . . . .	58
The Constant Table . . . . .	59
SDK Updates . . . . .	61
Conclusion . . . . .	61
Acknowledgments . . . . .	61

**Introduction to the vs\_3\_0 and ps\_3\_0 Shader Models 63**

Nicolas Thibieroz, Kristof Beets, and Aaron Burton

Introduction . . . . .	63
Features Common to vs_3_0 and ps_3_0 . . . . .	64
Flexible Input and Output Declarations . . . . .	64
Predication . . . . .	65
Static and Dynamic Flow Control . . . . .	66
Arbitrary Swizzle . . . . .	69
Destination Write Masks on Texture Instructions . . . . .	70
vs_3_0 Features . . . . .	71
Registers . . . . .	71
Instructions . . . . .	73
Texture Sampling . . . . .	73
Vertex Stream Frequency . . . . .	76
ps_3_0 Features . . . . .	78
Registers . . . . .	78
Instructions . . . . .	80
Unlimited Texture Samples and Dependent Reads . . . . .	82
Conclusion . . . . .	82
References . . . . .	82

**Advanced Lighting and Shading with Direct3D 9 83**

Michal Valient

Introduction . . . . .	83
Per-Pixel Phong . . . . .	84
Phong's Lighting Equation . . . . .	84
Vertex and Pixel Shaders 2.0 . . . . .	85
Vertex and Pixel Shaders 3.0 . . . . .	97
Per-pixel Environment Bump Mapping with Fresnel Term . . . . .	108
Mathematical Background . . . . .	109

Vertex Shader . . . . .	112
Pixel Shader 1.4 . . . . .	115
Pixel Shader 2.0 . . . . .	117
HLSL Version . . . . .	119
Background for Advanced Models . . . . .	122
Spherical Coordinates . . . . .	122
Roughness of a Surface . . . . .	123
Masking and Shadowing . . . . .	124
The Oren-Nayar Model . . . . .	125
Shaders . . . . .	127
HLSL Version . . . . .	131
Cook-Torrance Model . . . . .	134
Shaders 2.0 . . . . .	136
Shaders 1.4 . . . . .	140
HLSL Version . . . . .	143
Quality Comparison . . . . .	147
Conclusion . . . . .	148
References . . . . .	149

**Introduction to Different Fog Effects 151**

Markus Nuebel

Introduction . . . . .	151
The Theory behind Fog Calculations . . . . .	152
Technique One: Linear Fog . . . . .	154
Fog Equation . . . . .	154
Implementation. . . . .	155
Technique Two: Exponential Fog . . . . .	157
Fog Equation . . . . .	158
Implementation. . . . .	159
Technique Three: Exponential Squared Fog. . . . .	162
Fog Equation . . . . .	163
Implementation. . . . .	164
Technique Four: Layered Fog . . . . .	166
Theory and Equations. . . . .	167
Implementation. . . . .	168
Technique Five: Animated Fog . . . . .	174
Theory and Equations. . . . .	175
Implementation. . . . .	176
Conclusion . . . . .	178
References . . . . .	179

**Shadow Mapping with Direct3D 9 181**

Michal Valient

Introduction . . . . .	181
Shadow Algorithm. . . . .	182

Depth Bias Problem . . . . .	183
Shadow Map Filtering . . . . .	185
Shaders for Shadow Map Creation. . . . .	187
Shaders for Final Rendering . . . . .	188
Conclusion . . . . .	194
References . . . . .	195

**The Theory of Stencil Shadow Volumes 197**

Hun Yen Kwoon

Introduction . . . . .	197
Shadow Volume Concept . . . . .	199
Depth-pass (z-pass). . . . .	201
Depth-fail (z-fail) . . . . .	205
Problems and Solutions . . . . .	209
Finite Shadow Cover . . . . .	209
Ghost Shadow . . . . .	210
View Frustum Clipping . . . . .	212
Implementation on CPU . . . . .	220
How It Is Done . . . . .	220
Silhouette Determination . . . . .	221
Forming the Shadow Volume . . . . .	225
Shadow Volume Capping. . . . .	231
Depth-pass Stenciling Operations (DepthPassCPU). . . . .	233
Depth-fail Stenciling Operations (DepthFailCPU). . . . .	238
Rendering Shadow Volume Capping . . . . .	241
Implementation on GPU (Shaders) . . . . .	243
How It Is Done . . . . .	244
Preprocessing of Data . . . . .	245
Forming Shadow Volume in Shaders . . . . .	249
Vertex Shader Implementation (FiniteGPU). . . . .	250
Vertex Shader Implementation (InfiniteGPU) . . . . .	256
Better with Shaders? . . . . .	260
DirectX 9 HLSL Samples . . . . .	262
Efficiency and Robustness . . . . .	267
Use Less for More . . . . .	267
Cheat Whenever You Can . . . . .	269
Fighting the Invisible . . . . .	270
Scene Management Inside and Out . . . . .	271
Always a Good Switch . . . . .	275
Mix and Match . . . . .	275
The End. . . . .	275
References . . . . .	276

<b>Shader Development Using RenderMonkey</b>	<b>279</b>
Natalya Tatarchuk	
Introduction . . . . .	279
Overview of the IDE . . . . .	281
Creation of Basic Illumination Effect . . . . .	282
Run-Time Database Overview . . . . .	283
Workspace View . . . . .	285
Variable Creation and Management . . . . .	286
Predefined RenderMonkey Variables . . . . .	288
Stream Mapping Module . . . . .	290
Model Management . . . . .	293
Managing Effects . . . . .	294
Pixel and Vertex Shaders . . . . .	295
Editing Shaders . . . . .	296
Vertex Shader Setup and Editing . . . . .	298
Compiling Your Shaders . . . . .	302
Output Window . . . . .	302
Shader Assembly or Compilation Errors . . . . .	302
Editing Assembly . . . . .	303
Pixel Shader Setup and Editing . . . . .	306
Preview Window . . . . .	308
Editing Variables . . . . .	310
Render State Block Management . . . . .	314
Texturing in RenderMonkey . . . . .	317
Texture Objects . . . . .	318
Using Textures with HLSL Shaders . . . . .	322
Rendering to a Texture . . . . .	324
Render Passes . . . . .	324
Renderable Texture Support . . . . .	325
Editing a Renderable Texture . . . . .	331
Editing a Render Target . . . . .	332
Artist Editor . . . . .	332
Editing Variables in the Artist Editor Module . . . . .	334
Summary . . . . .	337
<b>Tips for Creating Shader-Friendly 3D Models</b>	<b>339</b>
Gim Guan Chua	
Generating Suitable Texture Coordinates . . . . .	340
The Influence of "Vertex Weight" . . . . .	341
Problems with Non-Convex Surfaces . . . . .	343
Conclusion . . . . .	345
<b>Index</b>	<b>347</b>