Game Engine Architecture

Jason Gregory

A K Peters, Ltd.
Wellesley, Massachusetts

Contents

Foreword xiii

Preface xvii

I Foundations 1

1 Introduction 3

1.1 Structure of a Typical Game Team 5
1.2 What Is a Game? 8
1.3 What Is a Game Engine? 11
1.4 Engine Differences Across Genres 13
1.5 Game Engine Survey 25
1.6 Runtime Engine Architecture 28
1.7 Tools and the Asset Pipeline 49

2 Tools of the Trade 57

2.1 Version Control 57
2.2 Microsoft Visual Studio 66
2.3 Profiling Tools 85

2.4 Memory Leak and Corruption Detection 87

2.5 Other Tools 88

3 Fundamentals of Software Engineering for Games 91

3.1 C++ Review and Best Practices 91

3.2 Data, Code, and Memory in C/C++ 98

3.3 Catching and Handling Errors 128

4 3D Math for Games 137

4.1 Solving 3D Problems in 2D 137

4.2 Points and Vectors 138

4.3 Matrices 151

4.4 Quaternions 169

4.5 Comparison of Rotational Representations 177

4.6 Other Useful Mathematical Objects 181

4.7 Hardware-Accelerated SIMD Math 185

4.8 Random Number Generation 192

II Low-Level Engine Systems 195

5 Engine Support Systems 197

5.1 Subsystem Start-Up and Shut-Down 197

5.2 Memory Management 205

5.3 Containers 223

5.4 Strings 242

5.5 Engine Configuration 252

6 Resources and the File System 261

6.1 File System 262

6.2 The Resource Manager 272

7 The Game Loop and Real-Time Simulation 303

7.1 The Rendering Loop 303

7.2 The Game Loop 304

7.3 Game Loop Architectural Styles 307
7.4 Abstract Timelines 310
7.5 Measuring and Dealing with Time 312
7.6 Multiprocessor Game Loops 324
7.7 Networked Multiplayer Game Loops 333

8 Human Interface Devices (HID) 339

8.1 Types of Human Interface Devices 339
8.2 Interfacing with a HID 341
8.3 Types of Inputs 343
8.4 Types of Outputs 348
8.5 Game Engine HID Systems 349
8.6 Human Interface Devices in Practice 366

9 Tools for Debugging and Development 367

9.1 Logging and Tracing 367
9.2 Debug Drawing Facilities 372
9.3 In-Game Menus 379
9.4 In-Game Console 382
9.5 Debug Cameras and Pausing the Game 383
9.6 Cheats 384
9.7 Screen Shots and Movie Capture 384
9.8 In-Game Profiling 385

III Graphics and Motion 397

10 The Rendering Engine 399

10.1 Foundations of Depth-Buffered Triangle Rasterization 400
10.2 The Rendering Pipeline 444
10.3 Advanced Lighting and Global Illumination 469
10.4 Visual Effects and Overlays 481

11 Animation Systems 491

11.1 Types of Character Animation 491
11.2 Skeletons 496

11.3 Poses 499

11.4 Clips 504

11.5 Skinning and Matrix Palette Generation 518

11.6 Animation Blending 523

11.7 Post-Processing 542

11.8 Compression Techniques 545

11.9 Animation System Architecture 552

11.10 The Animation Pipeline 553

11.11 Action State Machines 568

11.12 Animation Controllers 593

12 Collision and Rigid Body Dynamics 595

12.1 Do You Want Physics in Your Game? 596

12.2 Collision/Physics Middleware 601

12.3 The Collision Detection System 603

12.4 Rigid Body Dynamics 630

12.5 Integrating a Physics Engine into Your Game 666

12.6 A Look Ahead: Advanced Physics Features 684

IV Gameplay 687

13 Introduction to Gameplay Systems 689

13.1 Anatomy of a Game World 690

13.2 Implementing Dynamic Elements: Game Objects 695

13.3 Data-Driven Game Engines 698

13.4 The Game World Editor 699

14 Runtime Gameplay Foundation Systems 711

14.1 Components of the Gameplay Foundation System 711

14.2 Runtime Object Model Architectures 715

14.3 World Chunk Data Formats 734

14.4 Loading and Streaming Game Worlds 741

14.5 Object References and World Queries 750

14.6 Updating Game Objects in Real Time 757

14.7 Events and Message-Passing 773
14.8 Scripting 794
14.9 High-Level Game Flow 817

V Conclusion 819

15 You Mean There's More? 821
15.1 Some Engine Systems We Didn't Cover 821
15.2 Gameplay Systems 823

References 827

Index 831