

Contents

1	Introduction	1
1.1	Software Visualization and Program Behavior	1
1.2	Types of Software Visualization Tools	2
1.3	Virtual Environments	3
1.4	Scope of This Book	3
1.5	Contributions	5
	References	6

Part I Software Visualization

2	Visualization Principles and Techniques	9
2.1	Graphic Design Principles	10
2.1.1	Show the Information	10
2.1.2	Maximize Information Density	10
2.1.3	Remove Useless Information	11
2.1.4	Remove Redundant Information	12
2.1.5	Iterate	12
2.2	Visualization Principles	13
2.2.1	Animation	13
2.2.2	Least Astonishment	14
2.2.3	Visual Metaphors	14
2.2.4	Interconnection	14
2.2.5	Interaction	15
2.2.6	Dynamic Scale	15
2.2.7	Static Backdrop	17
2.3	Visualization Techniques	17
2.3.1	Incremental Algorithms	18
2.3.2	Radial Coordinates	18
2.3.3	Mapping Data to Colors, Textures, and Shapes	19
	References	21

3	Software Instrumentation and Data Collection	23
3.1	The Role of Static Information.....	23
3.2	Log Files and Real-Time Event Monitoring	24
3.3	Inventory of Architecture Components.....	25
3.4	Standard Execution Monitoring Scenario	26
3.4.1	Sources of Relevant Execution Behavior.....	27
3.4.2	Selecting or Developing Appropriate Monitors	28
3.4.3	Running the Target Program	28
3.5	Framework Characteristics.....	29
3.5.1	Multitasking	29
3.5.2	Independence.....	30
3.5.3	Access.....	30
3.5.4	Multiple Monitors and Monitor Coordinators.....	30
3.5.5	Execution Control	31
3.5.6	Visualization Support	33
3.5.7	Simplified Graphics Programming	33
3.5.8	Support for Dual Input Streams	33
3.5.9	Inter-task Access Functions.....	34
3.6	Events.....	35
3.6.1	Event Codes and Values	35
3.6.2	Event Reporting and Masking.....	36
3.6.3	Obtaining Events Using evinit	37
3.6.4	Setting Up an Event Stream	37
3.6.5	EvGet().....	37
3.6.6	Event Masks, and Value Masks.....	38
3.6.7	Artificial Events	39
3.7	Instrumentation in the Unicon Runtime System	40
3.7.1	Explicit Source-Related Events.....	40
3.7.2	Implicit Runtime System Events.....	40
3.8	Anatomy of an Execution Monitor.....	41
3.8.1	Handling User Input.....	42
3.8.2	Querying the Target Program for more Information	42
3.8.3	2D Graphics Capabilities	43
3.8.4	Some Useful Library Procedures	44
3.9	Typical Evolution of a Visualization Tool.....	45
3.9.1	Generate Log Files	45
3.9.2	Depict the Log Files	45
3.9.3	Scale to Handle Real Problems	46
3.9.4	Focus on Behaviors of Interest.....	46
3.9.5	Add User-Directed Navigation.....	46
3.10	Instrumenting Code Written in Other Languages	46
	References	47

4 Visualizing Aspects of Program Behavior	49
4.1 Following the Locus of Execution	49
4.1.1 Location Events	50
4.1.2 A Simple Line Number Monitor	50
4.1.3 A Location Profile Scatterplot	51
4.1.4 Tracking Source File Changes	54
4.2 Monitoring Procedure Activity	55
4.2.1 Activation Trees	55
4.2.2 An Animated Call-Result Scatterplot	59
4.2.3 Algae	63
4.2.4 Maintaining the Current Source File	67
4.3 Monitoring Memory Usage	69
4.3.1 Allocation by Type	69
4.3.2 Cumulative Allocation by Type	72
4.3.3 Running Allocation by Type	75
4.3.4 Survival Rates Across Collections	77
4.4 Monitoring String Scanning	78
4.4.1 Overview of String Scanning	78
4.4.2 String Scanning Events	79
4.4.3 Absolute and Relative Position Changes	79
4.4.4 The Scanning Environment Tree	82
4.5 Monitoring Structure and Variable Usage	85
4.5.1 Visualizing Lists and List Accesses	85
4.5.2 Visualizing Heterogeneous Structures	89
4.5.3 Monitoring Variable References	92
References	96
5 Integrating Multiple Views	97
5.1 Monitor Coordinators	97
5.2 Eve: The Reference Monitor Coordinator	99
5.2.1 Computation of the Minimal Event Set	100
5.2.2 The Event Code Table	101
5.2.3 Event Handling	101
5.2.4 Eve's Main Loop	102
5.2.5 Interactive Error Conversion	103
5.3 Writing Monitors that Run Under a Monitor Coordinator	104
5.3.1 Client Environment	104
5.3.2 General-Purpose Artificial Events	104
5.3.3 Monitor Communication Example	105
5.4 Integrating Visualizations Using Subwindows	105
References	106
6 Sharing Visualizations Across a Network	107
6.1 Connections: Firewalls, Hole Punching, and Tunneling	107
6.1.1 Firewalls	108
6.1.2 Resorting to Existing Protocols	108

6.1.3	Hole Punching.....	108
6.1.4	Tunneling.....	109
6.2	Network Protocols and Message Types	109
6.2.1	Transmitting Strings and Structures	109
6.2.2	Serialization.....	110
6.2.3	Buffering	110
6.3	Messages that Convey Graphics Output	110
6.4	Messages for Events and Program Behavior	111
6.5	A Remote Memory Allocation Visualizer	112
 Part II Virtual Environments		
7	An Overview of Virtual Environments.....	115
7.1	Categories.....	115
7.2	Features and Usage Patterns.....	117
7.3	Users and Avatars.....	117
7.4	Collaborative Views	118
7.5	Virtual Objects.....	118
7.6	CVE: A Brief History	118
	References	119
 8	 Virtual Worlds Graphics and Modeling	 121
8.1	3D Graphics and Modeling 101	121
8.2	The Programming Language as a Graphics Engine.....	122
8.2.1	3D Graphics API	122
8.2.2	Improving 3D graphics performance	124
	References	125
 9	 Non-Player Characters and Quests	 127
9.1	NPCs	128
9.1.1	NPC Profiles.....	129
9.1.2	Knowledge Model.....	129
9.1.3	Dialogue Model.....	129
9.1.4	Behavior Model.....	130
9.2	Quests.....	131
9.2.1	Quest Repository.....	132
9.2.2	Quest Rating and User Reward via Peer Review.....	132
9.3	Design and Implementation	133
9.3.1	NPC Architecture	133
9.3.2	Implementation Discussion	134
	References	138
 10	 Dynamic Texturing in Virtual Environments.....	 139
10.1	Reloading Textures.....	139
10.2	Textures as Windows.....	140
10.3	Dynamic Texture Examples	141

Contents	xiii
11 Embedding Visualizations in a Virtual Environment	145
11.1 Placement of Visualizations in Predefined Locations	145
11.2 Distributed Clones	147
11.3 Control Flow and Time Warp	147
11.4 Bug Hunting	149
Appendices	
Appendix A: Event Codes	151
A.1 Classes of Events.....	151
A.2 Individual Events.....	152
Appendix B: Software and Supporting Documentation	155
References	155

Writing Virtual Environments for Software Visualization

Jeffery, C.; Al-Gharaibeh, J.

2015, XVII, 155 p. 65 illus., 24 illus. in color., Hardcover

ISBN: 978-1-4614-1754-5