

Contents

Preface	xi
Foreword	xiii
1 Introduction	1
1.1 Visual Languages and Programming	1
1.2 Visual Programming vs. Program Visualization	3
1.3 Organization of the Book	5
1.4 General Readings on Visual Languages	7
2 The Foundation - Graph Grammars	9
2.1 Introduction	9
2.2 A Case Study	11
2.2.1 Process Flow Diagrams	11
2.2.2 Graph Rewriting Rules	14
2.2.3 A Graph Grammar for Process Flow Diagrams	17
2.3 Formal Definitions	20
2.3.1 Preliminaries	20
2.3.2 Reserved Graph Grammar and Its Properties	22
2.4 Graph Parsing	24
2.4.1 A Parsing Algorithm	24
2.4.2 Selection-Free Grammars	26
2.4.3 Parsing Complexity	29
2.5 Improvements Over the Layered Graph Grammar	31

2.6 Summary	33
2.7 Related Work	34
 3 Spatial Specification	 37
3.1 Introduction.....	37
3.2 The Spatial Graph Grammar Formalism	38
3.3 Spatial Relationships and Representations	39
3.3.1 Direction	39
3.3.2 Distance	40
3.3.3 Topology	40
3.3.4 Alignment	41
3.3.5 Size	41
3.3.6 Event Driven.....	42
3.3.7 Syntax-Directed Computations	43
3.4 Formal Definitions	43
3.5 Graph Parsing	46
3.5.1 A Parsing Algorithm.....	46
3.5.2 Object Sequencing.....	47
3.5.3 A Sequencing Example	52
3.6 Complexity Analysis	54
3.7 Summary.....	56
3.8 Related Work	56
 4 Multimedia Authoring and Presentation	 59
4.1 Introduction.....	59
4.2 Adaptation to Context Changes	61
4.2.1 The Marking Scheme.....	61
4.2.2 Size Adaptation.....	62
4.2.3 Style Adaptation	63

4.3 Example 1: Adapting Sizes for PDA Displays	64
4.3.1 Original Web and Resulting PDA Presentations.....	65
4.3.2 Structural Transformation.....	67
4.3.3 Grammatical Specification	69
4.4 Example 2: Adapting Presentation Styles	73
4.4.1 A Presentation Style	73
4.4.2 Grammatical Specification	73
4.4.3 Adapting to An Alternative Style	77
4.5 System Architecture and Implementation	78
4.6 Summary.....	81
4.7 Related Work	83
5 Data Interoperation.....	87
5.1 Introduction.....	87
5.2 A Hierarchical Interoperable Framework.....	89
5.3 Interoperation at Instance Level.....	92
5.3.1 Source and Target Documents.....	94
5.3.2 Specifying Structures and Translation Rules	96
5.3.3 Automatic Validation and Translation	99
5.4 Model and Schema Specifications	100
5.4.1 Identifying Meta Primitives.....	100
5.4.2 Defining a Model.....	101
5.4.3 Constructing a Schema	102
5.4.4 Drawing an Instance Data.....	103
5.4.5 Customizing the Host Graph	103
5.4.6 Adapting Rules	104
5.4.7 Drawing an Instance	105
5.5 Model Management Operators	106
5.5.1 Hierarchical Operations.....	108
5.5.2 Graphical Representation of Models and Mappings	109
5.5.3 Implementing Operators by Graph Transformation.....	111
5.5.4 Merge Operator.....	112
5.5.5 ModelGen Operator	113

5.5.6 Generalization of Operators	115
5.5.7 A Parsing Example	116
5.6 Summary	117
5.7 Related Work	118
6 Software Architecture Design	121
6.1 Introduction	121
6.2 Designing Architectural Styles	122
6.2.1 Components and Connectors	122
6.2.2 Architectural Styles	123
6.3 Designing an Architecture	126
6.3.1 Toll-Gates	126
6.3.2 Designing a Toll-Gate System	127
6.4 Verifying UML Class Diagrams	129
6.4.1 Class Diagrams	129
6.4.2 Automatic Verification	133
6.5 Visualizing Design Patterns	134
6.6 Transforming Software Architectures	135
6.7 Summary	137
6.8 Related Work	138
7 Web Engineering	143
7.1 Introduction	143
7.2 The Human-Web Interface	145
7.3 Using the HWI Tool	148
7.4 Graphical Programming for Web Design	149
7.4.1 Web Graphs and Design Notations	150

7.4.2 Graph Operations.....	152
7.4.3 Web Graph Grammar	153
7.4.4 Information Filtering	154
7.4.5 Supporting Multi-version Web Sites	155
7.5 Web Reuse Through Reverse Engineering	157
7.5.1 Reverse Engineering Approach	157
7.5.2 Web Visualization Algorithm.....	160
7.5.3 An Example	162
7.6 Summary.....	164
7.7 Related Work	164
8 Visual Language Generation.....	167
8.1 Introduction.....	167
8.1.2 Why Automatic Generation?	167
8.1.3 A Generic Multi-Level Approach	169
8.2 Design Criteria for VPEs	170
8.2.1 Heterogeneous Visual Programming.....	171
8.2.2 Hierarchical Structure.....	172
8.3 Design Model	173
8.3.1 The MVC Framework	173
8.3.2 An Ideal Design Model.....	176
8.3.3 The VisPro Design Model	177
8.4 The VisPro Toolset.....	179
8.4.1 A Case Study	180
8.4.2 Visual Object Generator	181
8.4.3 Control Specification Generator.....	184
8.4.4 Rule Specification Generator	187
8.4.5 Implementation	191
8.5 A Case Study: Generating A Distributed Programming Environment.....	192
8.5.1 PEDS.....	192
8.5.2 Generation of PEDS Using VisPro.....	195
8.6 Summary.....	200

8.7 Related Work	201
9 Conclusions and Future Perspective.....	203
9.1 Conclusions.....	203
9.2 Future Perspective	205
Bibliography.....	209
Appendix A RGG+: A Generalization to the RGG.....	227
A.1 Introduction.....	227
A.2 Notations	228
A.3 The Generalization.....	229
A.3.1 Definition of a RGG+ and Its Language	230
A.3.2 Decidability.....	231
A.4 Graph Parsing	231
A.4.1 A Parsing Algorithm.....	232
A.4.2 Search for Redexes	233
A.5 Parsing Complexities	234
A.5.1 Time Complexity	234
A.5.2 Space Complexity	236
A.5.3 Optimization Considerations	237
A.6 Summary.....	239
Index	241



<http://www.springer.com/978-0-387-29813-9>

Visual Languages and Applications

Zhang, K.

2007, XIV, 246 p. 109 illus., Hardcover

ISBN: 978-0-387-29813-9