

# **Reactive Internet Programming**

## ***State Chart XML in Action***

**Franck Barbier**

*University of Pau, France*

*ACM Books #10*

# Contents

i'refntr xiii

Advice for the Reader w  
Vocabulary \\  
Stylistic Conventions \\

## Chapter 1 Introduction 1

Software Engineering in the Internet Era 2  
Expected Benefits of Model-driven Software Development  
Programming with Events and States ↗  
Model Execution or Interpretation ↗  
Architectural Issues of Internet Programming 6

## Chapter 2 Event and State-based Modeling and Programming 1 >

States are Universal and Everywhere 1 s  
States are Abstract and Discrete 14  
"Event" as Dual Notion of State i >  
Harel's Statecharts I «  
Discovering State Chart XML 20  
Statechart Execution 2

## Chapter 3 Applying State Chart XML u

Startup of Barbados Crisis Management System ».)  
Business Case: "Route Negotiation" w  
Timing Constraints 1 ◊  
Introduction to the Run-to-completion Execution Mode s:  
Variations on Modeling: The Power of Statecharts Hi  
Evaluation !S

**Chapter 4 Programming State Chart XML Models >1**

- 1.1 Programming "My device" 7>2
- 1.2 Setup of Entry and Exit Actions 8
- 1.1.i Setup of Activities b)
- 1.1.i Sending Events Internally (>)
- 1.7 Action Parameters 70
- 1.1.f. State Machine Kick-off 72
- 1.1.r State Machine Shutdown 71
- 1.1.n State Machine Tracing 77.
- 1.1) Transition Programming 77
- 1.1.iii Guard Programming 77
- 1.1.i.i Event Processing 74
- 1.1.i) Communication Programming 80

**Chapter 5 Execution Semantics 87>**

- 5.1 Example of Execution Semantics Potential Defects 8f>
- "12 Run-to-Completion Cycles X7
- Action and Activity Execution Sequencing k(i
- 7.1 Execution Sequencing and Orthogonality 1)0
- Execution Sequencing and Nesting <)»
- >f Event Consumption Principle M1
- 7 Deferred Events <)1
- < s Transition Conflicts <»)

**Chapters Advanced Programming with PauWare engine 107**

- f>.i Completion Transitions 107
- i. ' Timer Services 112
- (p 7 State and Event Naming no
- 1. i Single Nesting 120
- i 7 The Notion of "internal transition" in State Chart XML 120
- i.i' Allowed Events at Large 121
- Unicast vs. Multicast Calls/Communications 12<>
- i ' C a c h e d Transitions vs. Contextual Transitions 12K
- i.7) Data Consistency Checking through State Invariants 1.i.i
- < .Hi History Facilities 157
- iv i i Concurrency 1 li)

**Chapter 7 Programming the "Internet of Things" 111**

- 1 Characterization of "Internet of Things" Applications 111
- ~ 2 Application Requirements (Home Automation System) 111
  - Analysis 117
  - 1 Design
  - Implementation Issues 119

**Chapter 8 Programming Web Enterprise Applications 181**

- k.1 Characterization of Web Enterprise Applications 181
- 8.2 Application Requirements (Railcar Control System) 181
  - s. i Analysis 186
  - ; > .i Design 187
  - s · Implementation Issues 189

**Chapter 9 Software Component Management 201**

- ↳ 1 Dynamical (Re)configuration 201
  - ' Extended Management Framework 211
- >>.i Pushing State Machine Data Outside 212

**Appendix A Internal Structure of PauWare Engine 211**

- \ i Organization of a State Machine's States in Memory 211
- \.2 Organization of a State Machine's Transitions in Memory 211

**Appendix B Acronyms 219****Appendix C Downloadable Software Resources 221**

- References 225
- Author's Bibliography 221