dandelon.com
© 2008 AGI-Information Management Consultants
May be used for personal purposes only or by
libraries associated to dandelon.com netwrk.

Ansgar Schleicher

Management of Development Processes

An Evolutionary Approach

With a Foreword by Prof. Dr.-Ing. Manfred Nagl

A 234996

Deutscher Universitäts-Verlag

Contents

1 Introduction 1

1.1 Process Management 2
1.1.1 Scope of Development Process Management 4
1.1.2 Benefits of Development Process Management 6
1.1.3 Characterization of Development Processes 7
1.1.4 Existing Systems for Process Management 8
1.1.5 Scope of this Book 13
1.2 Concepts and Practices for Process Management Systems 14
1.2.1 Layers of Modeling 15
1.2.2 System Architecture 22
1.3 The Challenge of Development Process Management 24
1.3.1 Dynamism in Development Processes 24
1.3.2 Vagueness in Process Management 25
1.3.3 Consequences 27
1.4 A Conceptual Framework for Process Evolution Support 28
1.4.1 Role of the Framework 29
1.4.2 Overview 29
1.4.3 Requirements on the Layers 31
1.4.4 New Relationships between the Layers 34
1.4.5 A Conceptual System Architecture 37
1.4.6 Relating Process Management Systems to the Framework 40
1.4.7 Relation to other Conceptual Frameworks 43
1.5 Contributions of this Book 47
1.6 Originality of the Approach 50

2 A Sample Development Process 53

2.1 A Technical Development Domain: Chemical Engineering 53
2.1.1 Scope of Chemical Engineering 53
2.1.2 Methods and Techniques 54
2.2 A Sample Process in Chemical Engineering 56
2.2.1 Polyamide 6 57
2.2.2 The Development Product 58
2.2.3 The Development Process 62
2.2.4 Characterization of the Sample Process 64

3 A Meta Model for Evolution Support : Dynamic Task Nets 67

3.1 Dynamic Task Nets by Example 68
3.1.1 Language Elements 68

3.1.2 Behavior of Dynamic Task Nets 71
3.1.3 The Polyamide 6 Process as a Dynamic Task Net 73
3.2 A Formal Meta Model 78
3.2.1 The Specification Language PROGRES 78
3.2.2 The Structural Base Model 80
3.2.3 The Dynamic Semantics 89
3.3 Support for Process Evolution 97
3.4 Related Process Meta Models 99
3.4.1 Process Programming 99
3.4.2 Attributed Grammars 99
3.4.3 Petri Nets 100
3.4.4 Rules 101
3.5 Summary 102

4 UML-based Process Model Definition 103

4.1 Requirements for Process Model Definition 103
4.2 Design Decisions 104
4.3 Mapping the DYNAMITE Meta Model onto UML 106
4.3.1 Motivation for Domain Meta Modeling 106
4.3.2 Evaluation Schema 110
4.3.3 Stereotypes as a Meta Modeling Construct 110
4.3.4 Meta Model Extensions 113
4.3.5 Comparison 116
4.3.6 The Meta Modeling Approach Followed in this Work 117
4.4 Structural Process Model Definition with the UML 118
4.4.1 Options for Structural Modeling in the UML 119
4.4.2 Class Diagrams for Type-level Process Model Definition 120
4.4.3 Collaboration Diagrams for Instance-level Process Model Definition 124
4.5 Behavioral Process Model Definition with the UML 127
4.5.1 Possibilities for Behavioral Process Model Definition 128
4.5.2 Behavioral Patterns in Detail 135
4.5.3 Collaboration Diagrams in Detail 140
4.6 Model Structuring 144
4.7 Related Approaches 145
4.7.1 Meta Modeling Approaches for the UML 145
4.7.2 Process Model Definition Languages 146
4.8 Summary 147

5 Interpreting Process Model Definitions 149

5.1 Overview 149
5.2 Transforming UML-based Process Model Definitions to PROGRES 151
5.2.1 Formalisms 152
5.2.2 Prerequisites for the Transformation 153

5.2.3 Transforming Class Diagrams 154
5.2.4 Transforming Collaboration Diagrams 163
5.3 On the Notion of Consistency 170
5.4 An Example 172
5.5 Propagation of Inconsistencies through a Task Net 175
5.6 Inconsistency Elimination 178
5.7 Formal Specification 181
5.7.1 The Consistency Checks 182
5.7.2 Specification of Consistency Checks 183
5.8 Discussion and Related Approaches 189
5.8.1 Real-World Inconsistencies 191
5.8.2 Model Inconsistencies 193
5.9 Summary 195

6 Infering Process Model Definition Knowledge 197

6.1 Scope 197
6.2 Knowledge Sources 199
6.2.1 Obtaining New Structural Knowledge 200
6.2.2 Obtaining New Behavioral Knowledge 201
6.3 An Inference Mechanism for Task Types 203
6.3.1 Prerequisites: A Semantical Product Model 204
6.3.2 Deriving Initial Task Types 205
6.3.3 An Algorithm for Affinity Calculation 206
6.3.4 Discussion of Abstract Examples 208
6.3.5 Clustering 209
6.3.6 Effects on the Process Model Definition Knowledge Base 210
6.3.7 Application to the Polyamide 6 Plant Development Process 211
6.4 An Inference Mechanism for Realization Types 214
6.4.1 Deriving Initial Realization Types 215
6.4.2 An Algorithm for Affinity Calculation 216
6.4.3 Discussion of Abstract Examples 219
6.4.4 Application to the Polyamide 6 Plant Development Process 220
6.5 Discussion 222
6.6 Related Approaches 224
6.6.1 Process Improvement 224
6.6.2 Process Model Inference 226
6.6.3 Database Administration Tools 226
6.6.4 Type Inference in Programming Languages 227
6.7 Summary 228

7 Process Model Definition Evolution and Task Net Migration 229

7.1 Scope 229
7.2 Requirements 230

7.3 Changing Process Model Definitions 231
7.3.1 State of the Art in Object-Oriented Database Management Systems 232
7.3.2 Conceptual Approach 234
7.3.3 Permitted Changes 235
7.3.4 Propagation of Changes through the Process Model Definition 237
7.3.5 An Example 239
7.3.6 Avoiding a Version Chaos 239
7.3.7 Effects on the Transformation to PROGRES 240
7.4 Task Net Migration 242
7.4.1 Possible Solutions 242
7.4.2 Using Inconsistency Toleration for Task Net Migration 244
7.4.3 Automated Migration 247
7.4.4 Formal Specification 248
7.5 Related Approaches in Process Management Systems 256
7.6 Summary 259

8 The Management System: Implementation and User Interface 261

8.1 Overview 261
8.2 Process Model Definitions in Rational Rose 263
8.2.1 A Design Overview 264
8.2.2 User Interface of the Process Model Definition Tools 269
8.3 Process Management Tools 268
8.3.1 Tool Generation 270
8.3.2 Design Overview 271
8.3.3 User Interface 277
8.4 The Inference Tool 280
8.4.1 Design Overview 280
8.4.2 User Interface 284
8.5 Summary 284

9 Conclusion 287

References 291