Contents

Foreword VII

Acknowledgements IX

Abstract XI

Contents XIII

List of Figures XIX
List of Tables XXIII
Abbreviations XXV

Introduction 1

Introduction to the Topic 1
Context of this Thesis Project 4
General Remarks on this Thesis 7
Registered Trademarks 7
Argumentation 7
Citations 8
General Structure of this Thesis 9

I. General Context and Theories 11

I.1 The *Model* Concept 13
I.2 Embedded Systems Development 16
I.2.1 Definition and Context 16
I.2.2 Characteristics 16
I.2.3 Embedded Development in the Automotive Domain 19
I.3 Software Engineering (SE) 24
I.4 Systems Engineering (SysEng) 26
I.5 Requirements Engineering and Management 31
I.5.1 The Term 'Requirement' 34
I.5.2 Phases, Artifacts and Techniques in *REM* 40
I.5.3 Requirements Management 43
I.5.4 Models in REM 44

Bibliografische Informationen
http://d-nb.info/1021587087

digitalisiert durch DEUTSCHE NATIONAL BIBLIOTHEK

I.5.5 Separation between Requirements and Design 48
I.5.6 The Role and Nature of Requirement Change........................ 49
I.5.7 Traceability in the Context of Requirements Management 55
I.5.7.1 Traceability in Different Aspects of Development Activities 57
I.5.7.2 Traceability as an Issue of Quality 61
I.5.7.3 The Potential Uses of Traceability 62
I.5.8 Deficiencies of Today's REM Practices.................................... 64

I.6 Design in Systems and Software Development.............................. 65
I.6.1 Different Design Phases in SysEng and SE............................ 66
I.6.1.1 System Design 67
I.6.1.2 Software Architecture 67
I.6.1.3 Detailed Design.. 70
I.6.2 General Theories about Design 70
I.6.2.1 Design as Symbolic Information Processing 71
I.6.2.2 Design as Wicked Problems... 84
I.6.2.3 Design as Situated Action 89
I.6.2.4 Design as a Language of Patterns 94
I.6.3 Comparison of General Design Theories 103
I.6.4 Dependency between Design Models and Code 105
I.6.5 Architecture Documentation... 107
I.6.6 Design in the Automotive Domain 110
I.6.6.1 Modeling Methods and Tools Used in Automotive Design 111
I.6.6.2 Integrating other Organizations into a Design 115

I.7 Quality Standards for *Safety-Critical* Development Processes....116
I.7.1 SPICE (ISO 15504)..119
I.7.1.1 The Process Reference Model of SPICE 120
I.7.1.2 The Measurement Framework 121
I.7.1.3 The Process Assessment Model (PAM) 122
I.7.2 Requirements, Design and Traceability in the Context of SPICE 124
I.7.2.1 ENG.1: Requirements Elicitation 124
I.7.2.2 ENG.2: System Requirements Analysis 126
I.7.2.3 ENG.3: System Architectural Design......................... 130
I.7.2.4 ENG.4: Software Requirements Analysis 132
I.7.2.5 ENG.5: Software Design.. 133
I.7.2.6 ENG.6: Software Construction.................................... 134
I.7.2.7 SUP.10: Change Management.................................... 135

I.7.3 Traceability in SPICE 137
I.7.3.1 Intersect: Dangers of Prescriptive Process Models ... 138
I.7.3.2 The Nature of the ENG-Processes, *Traceability*, and its Implications 142
I.7.4 Automotive SPICE 148
I.7.5 Safety Engineering: IEC 61508, ISO 26262 151
I.8 Feedback from Embedded Practice 153

II. Rationale Management and Traceability in Detailed Discussion 159

II.9 Rationale Management in Systems and Software Engineering 159
II.9.1 Characterization Criteria for Rationale Approaches 162
II.9.1.1 Representation 162
II.9.1.2 Basic Rationale Processes 163
II.9.1.3 Descriptive versus Prescriptive Approaches 164
II.9.1.4 Intrusiveness 164
II.9.2 Rationale Management Systems (RMS) 165
II.9.3 Overview of Different Rationale Approaches 166
II.9.3.1 Schemas for Argumentation 166
II.9.3.2 Approaches beyond Argumentation 173
II.9.3.3 Alternative Categorization 175
II.9.4 Why Rationale Management Could not yet Succeed in Practice 177
II.9.4.1 Cognitive Limitations 178
II.9.4.2 Rationale Capture Limitations as Central Challenge in Rationale Management 179
II.9.4.3 Retrieval Limitations 186
II.9.4.4 Usage Limitations 186
II.9.4.5 Synopsis of Rationale Limitations Concerning Alternative Design Theories 187
II.9.5 The Role of Rationale in System and Software Design ... 188
II.10 Requirements Traceability 192
II.10.1 Overview 192
II.10.2 Traceability and Consistency Gaps between Artifacts 194
II.10.3 Impact Analysis and Traceability 197
II.10.4 Core Dimensions for Characterization 201
II.10.4.1 Purpose 202

II.10.4.2 Conceptual Trace Model 204
II.10.4.3 Process 229
II.10.4.4 Tools 234
II.10.5 Traceability and its Benefit Problem 242
II.10.6 Traceability between Requirements and Design 245
II.10.6.1 Theoretic Research Results 245
II.10.6.2 Tool Couplings between REM- and Design Tools in Practice 248
II.10.7 Traceability between Requirements, Design and Code 254
II.10.8 Rationale Management and Traceability 257

III. PROVEtech:R2A – A Tool for Dedicated Requirements Traceability 259

III.11 Research Goals 261

III.12 Accompanying Case Study 265

III.13 Closing the Tool Gap 268

III.14 Closing the Gap between Requirements and Design 271

III.15 Abstraction Layers and Abstraction Nodes 272

III.16 Models Crossing Tool-Barriers 280
III.16.1 Insertion: Coupling Different *REM*- and *Modeling Tools* 280
III.16.2 Integrating Several *Modeling Tools* in a Single Model.... 281

III.17 Basic Support Features of R2A 284
III.17.1 Support for Collaborative Design Tasks 284
III.17.2 The Notes Mechanism 285
III.17.3 Extensibility: XML-Reporting and User Tagging 286
III.17.4 Unique Identifier Support for any Item in R2A 287
III.17.5 Evolutionary *Traceability* – Recording History and Baselines 287
III.17.6 The Properties Dialog 288

III.18 Requirements and Requirements *Traceability* 290
III.18.1 Managing Requirement Sources 290
III.18.2 Establishing Requirements Traceability 293
III.18.2.1 *Traceability* Operations in R2A 296
III.18.2.2 The Requirement Influence Scope (RIS) 299
III.18.2.3 Representing Requirement Contextual Data 302
III.18.2.4 The Requirement Dribble Process (RDP) 304
III.18.2.5 Overview over Navigation and Handling of Requirements Aspects in R2A 311

III.19 Taxonomy of Requiremental Items 313

III.20 Support for Capturing Decisions 316
III.20.1 Relation to Approaches of Rationale Management 319
III.20.2 Effects on the Traceability Model 322
III.20.3 Example How to Tame the Development Process Model of SPICE 324
III.20.4 Implementation of the Decision Model in R2A 326
III.20.5 Additional Support of the Decision Model for Designers 337
III.20.5.1 Patterns 338
III.20.5.2 Ensuring Adequate Realization of Design and Decisions 339
III.20.5.3 Support for Architecture Evaluation 339

III.21 Resource Allocation as a Special Decision Making Case 341
III.21.1 *Budgeted Resource Constraints* as further *Requiremental Items* 343
III.21.2 Advantages for Collaboration and Sharing Project Knowledge 345
III.21.2.1 Within Project Refinement 345
III.21.2.2 Communicating Information across Organizational Boundaries 346
III.21.2.3 Change Management 347
III.21.2.4 Different Views on the Same Problem 348
III.21.3 Representing Budgeted Resource Constraints in SysML. 349
III.21.4 Combining both *Decision Models* 351

III.22 Managing Changes and Consistency 352
III.22.1 Usage of Traces – Managing *Requiremental* Changes 353
III.22.1.1 Selective Tracing: Impact Analysis 353
III.22.1.2 Interactive Tracing: The *Model Browser* 357
III.22.1.3 Non-Guided Tracing: Additional Features for Fast Look-Up 358
III.22.2 Consistency Maintenance of Requirements, *Traceability* and Design 359

III.23 Aspects of Embedding R2A in a Process Environment 362
III.23.1 Avoiding Redundancies in *Supplier Management* 363
III.23.2 *Traceability* over Several Artifact Models without Redundancies 365
III.23.3 Decoupled Development of Requirement and Design Artifacts 368

III.24 Overall Architecture of R2A 370
III.24.1 General Architecture 370
III.24.2 The *Meta-Model* 372
III.24.3 Further Interfaces 376

IV. Synopsis 379

IV.25 Summary of the Achieved Research Results 379
IV.26 Perspectives for Further Research 385
IV.27 Conclusions 392

Bibliography 395
Index 435