

Requirements Engineering

Jeremy Dick · Elizabeth Hull
Ken Jackson

Requirements Engineering

 Springer

Jeremy Dick, BSc (Eng), ACGI, DPhil, DIC, MA
Costain Group PLC
Maidenhead
UK

Ken Jackson, BSc, MSc, MBCS
Retired
Alton
UK

Elizabeth Hull, BSc, PhD, CITP, FBCS
Professor Emerita of Computing Science
Ulster University
Newtownabbey, Co Antrim
UK

ISBN 978-3-319-61072-6 ISBN 978-3-319-61073-3 (eBook)
DOI 10.1007/978-3-319-61073-3

Library of Congress Control Number: 2017952348

© Springer International Publishing Switzerland 2017, corrected publication November 2017
Originally published in the series "Practitioner Series"

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

We would like to dedicate this book as follows:

*To my wife Yvonne,
to my children Sebastian, Timothy, Angus,
Robin and Felicity,
and to my grandchildren Henry, Ethan,
Teddy, Tristan and Imogen.*

Jeremy Dick

To my late parents John and Edna Hull.

Elizabeth Hull

*To my wife Chris,
to my children and their spouses Kate, Stef,
Andy, Amy and Pete,
and to my grandchildren Lizzie, Alice, Emily,
Annabel and Sophie.*

Ken Jackson

The original version of the book front matter was revised: The affiliation of the third author's name was corrected. The Erratum to the book front matter is available online at: https://doi.org/10.1007/978-3-319-61073-3_10

Preface to the Fourth Edition

We are grateful to the many students and practitioners who have purchased our book, and helped it survive through to this, the fourth edition.

Two of us, Elizabeth Hull and Ken Jackson, are now enjoying retirement, leaving Jeremy Dick to continue gaining experience in practical, tool-supported requirements engineering.

This fourth edition is driven by new insights gained from this ongoing experience, and also by the evolution of the tools referenced in the book.

In particular, we have:

- Strengthened the discussion about the problem and solution spaces by introducing the concept of “design agnosticism.”
- Added information about SysML, the systems engineering modelling extensions to UML.
- Reworked most of the diagrams that are models of some kind using Enterprise Architect from Sparx Systems.
- Provided new material on traceability in Chap. 7 gleaned from practical application.
- Rewritten the chapter on DOORS to base it on IBM Rational DOORS Next Generation.
- Swapped the order of Chaps. 8 and 9 so that the book ends with the management section.
- Added a “final word” to the new Chap. 9 about cultural change as the key to success.

We hope that readers will, as a result, find the content fresh for meeting today’s Systems Engineering changes.

May 2017

Jeremy Dick
Elizabeth Hull
Ken Jackson

Preface to the Third Edition

In our desire to keep the material in this book current, the main driver in creating a new edition has been to keep up with the latest release of DOORS. Since the publication of Edition 2, Telelogic—the developers of DOORS—has been acquired by IBM, and the tool has become part of the IBM/Rational stable. While the basic functions of the tool remain unchanged, the look-and-feel has advanced considerably. Therefore, Chap. 9 has been updated for DOORS version 9.2.

At the same time, we felt the need to provide a more explicit definition of requirements engineering. In searching the literature, we could not find a satisfactory definition, and we have addressed this in Chap. 1.

Apart from this, there is an expanded description of Product Family Management in Chap. 8, and a variety of small corrections throughout.

We hope our readers—students and practitioners—continue to find this to be a valuable book in advancing their understanding of the topic.

April 2010

Elizabeth Hull
Ken Jackson
Jeremy Dick

Preface to the Second Edition

This second edition follows quickly on the first edition and is an indication of how fast the subject is changing and developing. In the past 2 years there have been significant advances and these are reflected in this new edition.

Essentially, this is an update that places more emphasis on modelling by describing a greater range of approaches to system modelling. It introduces the UML2, which is the recent standard approved by the OMG. There is also an enhanced discussion on the relationship between requirements management and modelling, which relates well to the concept of rich traceability.

The chapter on the requirements management tool DOORS has been revised to use version 7 of the tool and this is complemented with examples taken from the DOORS/Analyst tool which demonstrates how the concepts of modelling can be captured and created within DOORS.

The text is still aimed at students and practitioners of systems engineering who are keen to gain knowledge of using requirements engineering for system development.

As before, a website supporting additional material is available at:

<http://www.requirementsengineer.uk>

June 2004

Elizabeth Hull
Ken Jackson
Jeremy Dick

Preface to the First Edition

Requirements engineering is common sense, but it is perceived to be difficult and is not well understood. For these reasons it is generally not very well done. The ever-increasing pressures on an organisation are often given as the main reasons for not introducing a more disciplined approach to requirements engineering, but its aim will be to do the job properly, so the task of the requirements engineer is to work out how best to help the organisation achieve its goal.

Systems engineering is critical in today's industry and requirements engineering is an important stage of that overall process. A good process is a key to requirements engineering—it determines how efficiently and rapidly products can be generated. This is particularly important in a global competitive market where the “time to market” and meeting stakeholder requirements are the key success factors.

Requirements engineering is also about management and hence issues in relation to requirements and management blend to show how requirements can be used to manage systems development.

The book is concerned with engineering requirements and how systems engineers may be helped to create better requirements. A generic process is presented which assists the reader in gaining a good understanding of the essence of requirements engineering. The process is then instantiated for the problem and solution domains of development. This book also addresses the concept of system modelling and presents various techniques and methods which are widely used. An important feature of the book is the presentation of approaches to traceability, the way in which it is captured and discusses metrics which can be derived from traceability. Finally, the book presents an overview of DOORS which is a tool for requirements management. A case study is used to illustrate the process presented in the book and the features of the tool.

This book should be read by those systems engineers (requirements engineers) in industry, who, being practitioners are keen to gain knowledge of using requirements engineering for system development. The book will also be of interest to final year undergraduate students in Computer Science, Software Engineering and Systems Engineering studying a course in Requirements Engineering and also to postgraduate research students in Computer Science or more generally in Engineering.

The approach taken in this book is based on current research in requirements engineering, however, it has not only taken the academic view but has also built substantially on current experience of working in industry to enable system engineers to manage requirements (and projects) more successfully. It provides a snapshot, in this rapidly evolving subject, of what we see as best practice in requirements engineering today.

May 2002

Elizabeth Hull
Ken Jackson
Jeremy Dick

Acknowledgements

Thanks are due to a number of individuals and organisations who helped in various ways:

Richard Stevens, who inspired us with his work on requirements management and who laid the foundation for the work in this book. He was the founder of Requirements Engineering Limited (later Quality Systems and Software Limited) which developed the DOORS tool.

Les Oliver (who worked for Astrium at the time) for his assistance in the development of statecharts for agreement, qualification and satisfaction.

Praxis Critical Systems (now Altran Praxis) for their initial concept of design justification which become *Rich Traceability*.

Keith Collyer, Jill Burnett and other colleagues of Telelogic Limited for their contributions to ideas presented in this book and for reviews, comments, suggestions and encouragement.

Our various employers such as the University of Ulster, Telelogic Limited, IBM, Integrate Systems Engineering Limited and SyntheSys Systems Engineers Limited and our customers over 20 years who have given us, explicitly and implicitly, the latitude and opportunity to develop the ideas, materials and examples of this book.

Contents

1	Introduction	1
1.1	Introduction to Requirements	1
1.2	Introduction to Systems Engineering	4
1.3	Defining Requirements Engineering	7
1.3.1	Definition of a Requirement	7
1.3.2	Definition of a Stakeholder	8
1.3.3	Definition of Requirements Engineering	9
1.4	Requirements and Quality	10
1.5	Requirements and the Lifecycle	11
1.6	Requirements Tracing	14
1.7	Requirements and Modelling	18
1.8	Requirements and Testing	20
1.9	Requirements in the Problem and Solution Domains	21
1.10	Requirements and Design Agnosticism	23
1.11	Requirements and Interfaces	29
1.12	How to Read this Book	30
2	A Generic Process for Requirements Engineering	33
2.1	Introduction	33
2.2	Developing Systems	33
2.3	Generic Process Context	36
2.3.1	Input Requirements and Derived Requirements	38
2.3.2	Acceptance Criteria and Qualification Strategy	38
2.4	Generic Process Introduction	40
2.4.1	Ideal Development	40
2.4.2	Development in the Context of Change	41
2.5	Generic Process Information Model	43
2.5.1	Information Classes	43
2.5.2	Agreement State	45
2.5.3	Qualification State	46

- 2.5.4 Satisfaction State 47
- 2.5.5 Information Model Constraints. 48
- 2.6 Generic Process Details 49
 - 2.6.1 Agree Requirements 49
 - 2.6.2 Analyse and Model 51
 - 2.6.3 Derive Requirements and Qualification Strategy. 52
- 2.7 Summary 56
- 3 System Modelling for Requirements Engineering 57**
 - 3.1 Introduction 57
 - 3.2 Representations for Requirements Engineering 58
 - 3.2.1 Data Flow Diagrams 58
 - 3.2.2 Entity-Relationship Diagrams. 64
 - 3.2.3 Statecharts 65
 - 3.2.4 Object-Oriented Approaches 66
 - 3.3 Methods. 68
 - 3.3.1 Viewpoint Methods 69
 - 3.3.2 Object-Oriented Methods. 78
 - 3.3.3 The UML Notation 81
 - 3.3.4 SysML 86
 - 3.3.5 Formal Methods 89
 - 3.4 Model-Based Systems Engineering. 91
 - 3.5 Modelling and Qualification 91
 - 3.6 Summary 92
- 4 Writing and Reviewing Requirements 93**
 - 4.1 Introduction 93
 - 4.2 Requirements for Requirements 94
 - 4.3 Structuring Requirements Documents 94
 - 4.4 Key Requirements 96
 - 4.5 Using Attributes 97
 - 4.5.1 Definition of a Requirement Expression. 97
 - 4.6 Ensuring Consistency Across Requirements 99
 - 4.7 Value of a Requirement 100
 - 4.8 The Language of Requirements. 101
 - 4.9 Requirement Boilerplates 103
 - 4.10 Granularity of Requirements. 105
 - 4.11 Requirements and Modelling 107
 - 4.12 Criteria for Writing Requirements Statements 107
 - 4.13 Guide to Writing Requirements. 109
 - 4.14 Summary 111
- 5 Requirements Engineering in the Problem Domain 113**
 - 5.1 What is the Problem Domain? 113
 - 5.2 Instantiating the Generic Process. 114
 - 5.3 Agree Requirements with Customer 116

- 5.4 Analyse & Model 116
 - 5.4.1 Identify Stakeholders 117
 - 5.4.2 Create Usage Model 118
 - 5.4.3 Scoping the System 122
- 5.5 Derive Requirements 122
 - 5.5.1 Define Structure. 122
 - 5.5.2 Capture Requirements 126
 - 5.5.3 Define Acceptance Criteria. 133
 - 5.5.4 Define Qualification Strategy 133
- 5.6 Summary 134
- 6 Requirements Engineering in the Solution Domain 135**
 - 6.1 What is the Solution Domain 135
 - 6.2 Engineering Requirements From Stakeholder Requirements
to System Requirements 137
 - 6.2.1 Producing the System Model 137
 - 6.2.2 Creating System Models to Derive System Requirements 139
 - 6.2.3 Banking Example 144
 - 6.2.4 Car Example 147
 - 6.2.5 Deriving Requirements From a System Model 152
 - 6.2.6 Agreeing the System Requirements with the Design Team 153
 - 6.3 Engineering Requirements From System Requirements
to Subsystems. 154
 - 6.3.1 Creating a System Architecture Model 154
 - 6.3.2 Deriving Requirements From an Architectural
Design Model 156
 - 6.4 Other Transformations Using a Design Architecture 156
 - 6.5 Summary 157
- 7 Advanced Traceability 159**
 - 7.1 Introduction 159
 - 7.2 Elementary Traceability 160
 - 7.3 Rationale 162
 - 7.4 Satisfaction Statements 162
 - 7.5 Rich Traceability 163
 - 7.6 Supporting Information and Evidence 167
 - 7.7 Sufficiency and Necessity 169
 - 7.8 Reviewing Traceability. 170
 - 7.9 The Language of Satisfaction Statements 170
 - 7.10 Rich Traceability Analysis 170
 - 7.11 Requirements Flow-down 172
 - 7.12 Rich Traceability for Qualification 173
 - 7.13 Implementing Rich Traceability 174
 - 7.13.1 Single-layer Rich Traceability 174
 - 7.13.2 Multi-layer Rich Traceability 174
 - 7.14 Design Documents 175

- 7.15 Metrics for Traceability 179
 - 7.15.1 Breadth 180
 - 7.15.2 Depth 181
 - 7.15.3 Growth 181
 - 7.15.4 Balance 182
 - 7.15.5 Latent Change 183
- 7.16 Summary 185
- 8 DOORS: A Tool to Manage Requirements. 187**
 - 8.1 Introduction 187
 - 8.2 The Case for Requirements Management 188
 - 8.3 DOORS Information Model 188
 - 8.4 The Project Area 189
 - 8.5 Modules as Documents. 191
 - 8.6 Artifacts 192
 - 8.7 Traceability Links. 194
 - 8.8 Views. 196
 - 8.8.1 Rows in Views 197
 - 8.8.2 Columns in Views 199
 - 8.9 Artifact Types and Attributes 201
 - 8.10 Import and Export 205
 - 8.11 Summary 206
- 9 Management Aspects of Requirements Engineering 207**
 - 9.1 Introduction to Management 207
 - 9.2 Requirements Management Problems 208
 - 9.2.1 Summary of Requirement Management Problems. 210
 - 9.3 Managing Requirements in an Acquisition Organisation 210
 - 9.3.1 Planning 210
 - 9.3.2 Monitoring 213
 - 9.3.3 Changes 213
 - 9.4 Supplier Organisations 215
 - 9.4.1 Bid Management 215
 - 9.4.2 Development 219
 - 9.5 Product Organisations. 221
 - 9.5.1 Planning 221
 - 9.5.2 Monitoring 225
 - 9.5.3 Changes 226
 - 9.6 Agile Development. 227
 - 9.7 Organisational Culture 228
 - 9.8 Summary 229
 - 9.8.1 Planning 229
 - 9.8.2 Monitoring 230
 - 9.8.3 Changes 230
 - 9.8.4 Organisational Culture 230
- Erratum to: Requirements Engineering E1**
- Bibliography 231**
- Index 235**