



Colin Hood  
Simon Wiedemann  
Stefan Fichtinger  
Urte Pautz

# Requirements Management

The Interface Between  
Requirements Development and  
All Other Systems Engineering Processes

 Springer

# Requirements Management

Colin Hood · Simon Wiedemann · Stefan Fichtinger  
Urte Pautz

# Requirements Management

The Interface Between  
Requirements Development and  
All Other Systems Engineering Processes

Colin Hood  
Simon Wiedemann  
Stefan Fichtinger  
Urte Pautz  
Keltenring 7  
82041 Oberhaching  
Germany  
Colin.Hood@Hood-Group.com  
Simon.Wiedemann@Hood-Group.com  
Stefan.Fichtinger@Hood-Group.com  
Urte.Pautz@Hood-Group.com

ISBN 978-3-540-47689-4

e-ISBN 978-3-540-68476-3

DOI 10.1007/978-3-540-68476-3

Library of Congress Control Number: 2007938804

© 2008 Springer-Verlag Berlin Heidelberg

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, 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.

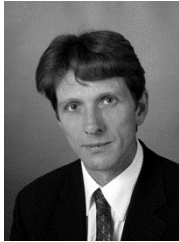
*Cover Design:* KüinkelLopka, Heidelberg

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

## The authors



Colin Hood is founder and Chairman of the HOOD Group. He has been responsible for the development of control systems from relay based systems to modern electronic and software controlled systems. Colin holds an BSc(Hons) in Electrical Engineering and Electronics, a Diploma in Management Studies (DMS), and an MBA. Colin is also a founding member of the International Requirements Engineering Board (IREB).



Simon Wiedemann has been a Senior Consultant with the HOOD Group for a couple of years and is now focussing on his responsibilities as a member of the HOOD Group's technical board, approving publications and training consultants. His PhD is on mathematical methods for numerical simulations of flexible multibody systems. Simon is a professor for mechanical engineering at the Munich University of Applied Sciences.



Stefan Fichtinger is a Senior Consultant with the HOOD Group. Since the beginning of his professional career, Stefan has been involved in the requirements definition of software products such as product data management (PDM) and logistics systems. As a software product manager, Stefan was responsible for market analyses and successful roll-outs.. He also has experience as process manager. Stefan holds a Dipl.-Ing. degree.



Urte Pautz is a Senior Consultant with the HOOD Group, supporting customers in introducing and establishing requirements management and engineering processes. To this end she uses assessments, seminars, workshops and coaching. Urte has many years of experience in configuration and change management, and also has a strong background in information technology and software development. Urte holds a Dipl.-Inf. degree.

To contact the authors of this book, please use the following email address:

*Technical-Board@HOOD-Group.com*

You are also welcome to visit the HOOD Group's homepage using:

*www.HOOD-Group.com*

## **Preface and Dedications**

We wrote this book to help people that have been trained in one discipline at the expense of achieving a balanced view of complete systems. We are systems engineers that are experts in requirements. We are systems engineers first and foremost, because without an appreciation of all of the disciplines of systems engineering we would not be able to appreciate the finer points of our speciality. Without an appreciation of all of the disciplines of systems engineering we could not be experts in our field.

We gratefully acknowledge the support we received from a team of people from the HOOD Group in the writing of this book. Amongst them, we specially want to say thank you to

Gabi Leibmann: she kept a lot of administrative work away from us

Stefan Fichtinger: for doing much more than has been agreed

Michael Jastram: for all his incredible support

Munich, October 2007  
Colin Hood

# Table of Contents

- 1 Introduction..... 1**
- 1.1 Aim of Book ..... 1
- 1.2 Benefit to be gained from book ..... 1
- 1.3 Definition of terms ..... 2
- 1.4 Structure of the Book ..... 4
- 1.5 How to read this book ..... 7
  
- 2 Why Requirements Management and Engineering..... 11**
- 2.1 General ..... 11
- 2.2 Advantages of RM&E in project management ..... 13
  - 2.2.1 Advantages of project planning ..... 13
  - 2.2.2 Advantages during the implementation phase ..... 14
  - 2.2.3 Advantages during the acceptance phase ..... 15
  - 2.2.4 Advantages in regard to version and configuration management ..... 15
- 2.3 Advantages for finding solutions in design and architecture ..... 16
  - 2.3.1 Advantages design and architecture ..... 16
  - 2.3.2 Advantages in finding solutions ..... 17
- 2.4 Advantages in purchase and supplier management ..... 18
- 2.5 Advantages in customer service, sales and marketing ..... 18
- 2.6 Advantages in test and verification management ..... 19
  
- 3 Processes and Methods in Requirements Management and Engineering..... 21**
- 3.1 The roots of Requirements Management and Engineering ..... 21
  - 3.1.1 The progress in computer engineering ..... 22
  - 3.1.2 Programmers – from artists to facilitators ..... 24
  - 3.1.3 Requirements Management and Engineering today ..... 27
- 3.2 Common concepts in Requirements Management and Engineering ..... 29
  - 3.2.1 The systems engineering concept ..... 29
  - 3.2.2 The requirements management concept ..... 30
  - 3.2.3 The process quality concept ..... 31
- 3.3 Processes and methods in Requirements Management and Engineering ..... 31
  - 3.3.1 Requirements engineering ..... 32
  - 3.3.2 Requirements management ..... 35
- 3.4 Summary ..... 38
  
- 4 Introduction to Requirements Engineering..... 39**
- 4.1 History of Requirements Engineering ..... 39
- 4.2 HOOD Requirements Definition Process ..... 42
  - 4.2.1 Aim of this part ..... 42
  - 4.2.2 What is a Process? ..... 43
  - 4.2.3 The activities of the HOOD Requirements Definition Process ..... 45
  - 4.2.4 Modelling ..... 45
- 4.3 Requirements Development and Requirements Engineering ..... 55
- 4.4 Summary ..... 57



<b>5</b>	<b>Introduction to Requirements Management.....</b>	<b>59</b>
5.1	What is Requirements Management .....	59
5.2	Why we need Requirements Management .....	59
5.3	The benefits of a working Requirements Management .....	65
5.4	Why some people are against Requirements Management .....	71
5.5	How resistance can be avoided .....	73
5.6	After the introduction of Requirement Management.....	76
5.7	Summary .....	77
<b>6</b>	<b>Project Management interface.....</b>	<b>79</b>
6.1	What is Project Management .....	79
6.2	How Requirements Management can support the writing of proposals .....	80
6.3	How Requirements Management can support the definition of the project scope .....	83
6.4	How Requirements Management can support estimating resources and costs .....	84
6.5	How Requirements Management can support project planning (scheduling) .....	85
6.6	How Requirements Management can support project monitoring.....	88
6.7	How Requirements Management can support quality management.....	90
6.8	How Requirements Management can support reporting.....	95
6.9	How Requirements Management can support managing people .....	96
6.10	Summary .....	98
<b>7</b>	<b>Configuration Management interface.....</b>	<b>101</b>
7.1	Of versions, configurations, and releases .....	102
7.2	Management Disciplines and the German Government V-Modell.....	107
7.3	Configurations in the Context of Requirements Management .....	108
7.3.1	Changes of requirements and specifications in practice.....	109
7.3.2	Requirements Management – Configuration Units .....	112
7.4	Traceability in Requirement Management and Configuration Management.....	114
7.5	Tool Use for Version and Configuration Management .....	115
7.5.1	Solution Concept:Traceability in Practice .....	116
7.6	Summary .....	117
<b>8</b>	<b>Metrics and Analysis.....</b>	<b>121</b>
8.1	Metrics – general .....	121
8.2	The Importance of Metrics.....	122
8.3	Attributes of Metrics .....	123
8.3.1	Goals Supported by the Metric .....	123
8.3.2	Customers of the Metric.....	124
8.3.3	Interval of Measurement .....	124
8.3.4	Measurements Used .....	124
8.3.5	Unit of Measurement.....	124
8.3.6	Data Source (Effort required to capture /reliability).....	124
8.3.7	Interpretation of Results.....	125
8.3.8	Strengths and Weaknesses of the Metric .....	125
8.3.9	Prerequisites for Measurement.....	125
8.3.10	Presentation Format for the Metric .....	125
8.4	Typical Improvement Goals with RM&E .....	126
8.4.1	Reduction in Change Costs .....	126
8.4.2	Reaching CMMI Level 3 in an Assessment .....	127
8.4.3	Reaching a Specific SPICE Level in an Assessment.....	127
8.4.4	Introducing and Establishing RM&E Methodology in Pilot Projects.....	128
8.4.5	Creating Basic Know-How in RM&E Amongst Employees .....	128
8.4.6	Improving the Quality of an RM&E Process.....	128
8.4.7	Improving Customer and Supplier Specifications .....	129
8.5	Example of a Metric.....	130
8.5.1	Creating a basic level of know-how in RM&E amongst staff.....	130
8.6	The Evaluation of a Metric by Management .....	132

8.7	Psychological Aspects of Introducing RM&E Metrics.....	133
8.8	Summary.....	135
<b>9</b>	<b>Risk Management interface.....</b>	<b>137</b>
9.1	What is a risk .....	137
9.2	What is Risk Management.....	138
9.3	Preparing a Risk Management.....	138
9.4	The Risk Management process.....	141
9.4.1	Risk identification and how Requirements Management can support.....	142
9.4.2	Risk assessment and how Requirements Management can support.....	148
9.4.3	Definition of countermeasures and how Requirements Management can support .....	151
9.4.4	Monitoring risks and how Requirements Management can support .....	153
9.5	Summary.....	154
<b>10</b>	<b>Test Management (Validation and Verification) interface .....</b>	<b>157</b>
10.1	What are Validation and Verification?.....	157
10.2	The Validation and Verification planning process.....	158
10.3	The role of Requirements Management in Validation and Verification .....	160
10.3.1	Requirements Management supports in defining the test scope .....	160
10.3.2	Requirements Management supports in documenting the test method .....	164
10.3.3	Requirements Management supports in documenting who carries out the verification.....	166
10.3.4	Requirements Management supports in defining when to carry out verifications .....	168
10.3.5	Requirements Management supports in estimating the costs of verification .....	171
10.3.6	Requirements Management supports in estimating the effort needed for verification.....	172
10.4	Summary.....	173
<b>11</b>	<b>Change Management interface .....</b>	<b>175</b>
11.1	General.....	175
11.2	Basics of Change Management .....	175
11.3	Factors Influencing Change.....	176
11.4	Number of Changes during Development.....	177
11.5	Two Phases of Change Management: Informing and Approval-based .....	178
11.5.1	Informing Change Management.....	179
11.5.2	Approving Change Management.....	180
11.6	Turning Change Management theory into practice.....	186
11.6.1	Effects of a Lack of Change Management .....	187
11.6.2	Management support for introducing processes.....	188
11.7	Procedure for Introducing Structured Change Management .....	189
11.8	Summary.....	191
<b>12</b>	<b>Advanced Requirements Management: the complete specification.....</b>	<b>193</b>
12.1	Interfaces between other Systems Engineering disciplines and Requirements .....	193
12.2	Getting away from the document view .....	195
12.2.1	The document view.....	195
12.2.2	The information view .....	196
12.3	Implementing Requirements Management .....	198
12.3.1	Implementing the interface to Project Management and Quality Management .....	200
12.3.2	Implementing the interface to Version Management and Configuration Management.....	203
12.3.3	Implementing the interface to Risk Management .....	207
12.3.4	Implementing the interface to Test Management.....	208
12.3.5	Implementing the interface to Change Management .....	209
12.3.6	Overview.....	212
12.4	Summary.....	213

- 13 The HOOD Capability Models ..... 215**
- 13.1 The meaning of capability models ..... 215
- 13.2 Why we need capability models ..... 216
- 13.3 Two example capability models ..... 218
  - 13.3.1 SPICE ..... 218
  - 13.3.2 CMMI ..... 220
- 13.4 HOOD Capability Model for Requirements Definition ..... 221
- 13.5 HOOD Capability Model for Requirements Management ..... 222
- 13.6 Summary ..... 222
  
- 14 The HOOD Capability Model for Requirements Definition ..... 223**
- 14.1 Brief repetition of the HOOD Requirements Definition Process ..... 223
- 14.2 The idea behind the HOOD capability model for requirements definition ..... 224
- 14.3 The structure of the HOOD capability model for requirements definition ..... 226
- 14.4 How to use the HOOD capability model for requirements definition ..... 228
  - 14.4.1 Level 1: Getting started ..... 228
  - 14.4.2 Level 2: Capable ..... 236
  - 14.4.3 Level 3: Expert ..... 239
- 14.5 Summary ..... 242
  
- 15 The HOOD Capability Model for Requirements Management ..... 243**
- 15.1 The structure of the HOOD capability model for requirements management ..... 243
- 15.2 How to use the HOOD capability model for requirements management ..... 244
  - 15.2.1 Level 1: Getting started ..... 245
  - 15.2.2 Level 2: Capable ..... 252
  - 15.2.3 Level 3: Expert ..... 259
- 15.3 Summary ..... 265
  
- List of References ..... 267**
  
- Index ..... 273**

# 1 Introduction

## 1.1 Aim of Book

The aims of this book are to motivate successful improvements to requirements management, to promote understanding of requirements management as one of an interrelated set of systems engineering disciplines, and to understand these systems engineering disciplines and their interfaces to requirements processes.

## 1.2 Benefit to be gained from book

By understanding and following the guidance in this book you will be able to reap benefits of synergy between team members and across departmental boundaries by coordinating efforts in requirements management as part of your systems engineering activities.

We have seen organisations that as they have grown have developed to become a collection of independent departments. Too often these departments concentrate increasingly on achieving their departmental aims, eventually to the extent that their departmental aims become more important than the aims of the overall organisation. What we need is coordinated teamwork where each part of the team pulls in the same direction.

This book helps a team to understand the central role played by requirements in systems engineering projects. It shows that no one systems engineering discipline is more important than any other. It shows that all the systems engineering disciplines are interrelated and interdependent.

This book will establish the need for, and legitimise the use of requirements management and engineering.

Managing requirements consists of managing changes to requirements, managing various versions of requirements, managing multiple configurations of requirements, managing deliveries of requirements on time, in budget and to the correct quality without taking undue risks. And

all the time ensuring that all those who need to know, know who is responsible for what. All of this requires communication and commitment.

Product and services produced to meet requirements must be checked against requirements to ensure that the specified and agreed requirements have been achieved

A perfectly optimized system is a set of suboptimal subsystems. If teams try to optimise each subsystem there will be conflict. Following the advice in this book teams will be inspired to see the big picture and be able to concentrate on getting the system built as required.

To introduce terms such as RM&E (requirements management and engineering) and relate to other nomenclature so that CMMI (Capability Maturity Model Integration) terms may be used throughout the book

### **1.3 Definition of terms**

**CMMI:** Capability Maturity Model Integration. A framework for scoring an organisation's ability to work with systems engineering processes. CMMI comes from the Software Engineering Institute (SEI) of Carnegie Mellon University in Pittsburgh U.S.A. Various trademarks and service marks of the SEI relating to CMMI are acknowledged.

**HCM:** HOOD capability model. A model for judging the quality of the implementation of a process mainly by considering the quality of its work products. Often used to support motivation of change programmes by measuring and publishing progress.

**Process:** (see also Software Process). A sequence of steps performed for a given purpose; for example, the software development process.

**Requirement:** A statement identifying a capability, physical characteristic, or quality factor that bounds a product or process need for which a solution will be pursued.

**Requirements Definition:** The process of producing documented and agreed requirements by means of elicitation, specification, analysis (quality check: judgment of requirements against quality criteria), and review (leading to acceptance, rejection, or return for rework) of requirements.

**Requirements Development:** The purpose of requirements development is to produce and analyze customer, product, and product-component requirements.

Requirements Engineering: See Requirements Development.

Requirements Management: The set of procedures that support the development of requirements including planning, traceability, impact analysis, change management and so on.

Requirements Management: The sum of the interfaces between requirements development and all other systems engineering disciplines such as configuration management and project management. The purpose of requirements management is to manage the requirements of the project's products and product components and to identify inconsistencies between those requirements and the project's plans and work products.

RM&E: Requirements management and engineering. The overall term used to include all requirements related processes.

Note to RM&E: In the 1990's the overall term used was requirements management. Then towards the end of the 1990's and early in the new millennium a trend gathered momentum to split the management of requirements from the development of requirements. Some organizations made the distinction along the lines that developing or defining requirements was requirements engineering. Others disagree. Some organisations use both terms requirements management and requirements engineering and consider that their understanding is the one and only true definition. Other organisations use definitions that completely contradict the understanding of others, and also consider that their understanding is the one and only true definition. Some use the CMMI definitions of requirements management and requirements development and combine these by using requirements engineering to encompass everything. So we use the term requirements management and engineering in an attempt to include all people, while acknowledging that there are a variety of definitions. At work we use whatever terminology our customers wish. There are more important battles to fight than who has the best words. People who get hung up on whose definition is correct, (or more normally the fight is who is incorrect!) should read A. A .Milne and learn from Winnie the Pooh; "We can use words to mean whatever we wish them to mean". As long as we understand each other we can work together. We advise the use of standards wherever possible. Where there is no single standard we must agree amongst ourselves.

**Software Process:** (see also Process). A set of activities, methods, practices, and transformations that people use to develop and maintain software and the associated products.

**Stakeholder:** A “stakeholder” is a group or individual that is affected by or in some way accountable for the outcome of an undertaking. Stakeholders may include project members, suppliers, customers, end users, and others.

**Alternative definition:** People who will be affected by the project or can influence it but who are not directly involved with doing the project work. Examples are managers affected by the project, process owners, people who work with the process under study, internal departments that support the process, customers, suppliers, and financial department.

**Alternative definition:** People who are (or might be) affected by any action taken by an organization. Examples are: customers, owners, employees, associates, partners, contractors, suppliers, related people or located near by.

**Alternative definition:** Any group or individual who can affect or who is affected by achievement of a firm's objectives

**Test:** (See Validation and Verification). The activity of checking correctness.

**Verification:** Although “verification” and “validation” at first seem quite similar in CMMI models, on closer inspection you can see that each addresses different issues. Verification confirms that work products properly reflect the requirements specified for them. In other words, verification ensures that “you built it right.”

**Validation:** Validation confirms that the product, as provided, will fulfill its intended use. In other words, validation ensures that “you built the right thing.”

## **1.4 Structure of the Book**

This book is divided into three parts. Part one is Requirements Management and Engineering: requirements management is greater than