Requirements Management

Colin Hood · Simon Wiedemann · Stefan Fichtinger
Urte Pautz

Requirements Management

The Interface Between Requirements Development and All Other Systems Engineering Processes

Springer

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

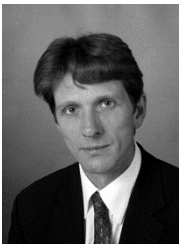
Cover Design: KünkelLopka, Heidelberg

Printed on acid-free paper

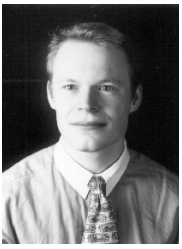
9 8 7 6 5 4 3 2 1

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 Engineerig 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 mangement 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 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 that 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 Melon 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

the sum of its parts. In this first part the aim is to introduce the book and to establish a common understanding of terminology with the reader.

Part two is Getting Down and Dirty: the low-down on systems engineering disciplines and their interfaces to requirements processes. In this part the aim is to discuss in detail the systems engineering disciplines, and specifically to define the interface between each systems engineering discipline and requirements development.

Part three is A Practical guide: helping to motivate and support successful implementation of requirements driven improvements with HOOD capability models (HCM).

Part one starts with Chapter 1 Introduction. This is the introduction to the book and describes the aims of the book and the benefits to be gained from reading the book. Some terms are defined that aid the understanding of the following chapters. Some terms are defined more than once quoting from various sources to show that the terms, although used often, are not standardized. If these definitions are contradictory it is made clear which definition is to be used in this book. The structure of the book is explained and a guide is given how to read this book to get the best advantage.

Chapter 2 Why Requirements Management and Engineering? In this chapter the need for and benefits of requirements management and engineering (RM&E) are explained. Terms such as RM&E and other nomenclature are explained so that CMMI terms may be used throughout the book. Definitions and detailed introduction of interfaces between systems engineering disciplines is not done here, but are investigated later in chapter 5.

Chapter 3 is Processes and Methods in Requirements Management and Engineering. The aim is to introduce the other systems engineering disciplines and process including those described by CMMI which will be explained in detail to form the structure of Part 2, the main part of this book.

Chapter 4 is Introduction to Requirements Engineering. The chapter introduces and defines requirements engineering, requirements development, and the HOOD requirements definition process.

Chapter 5 is Introduction to Requirements Management. Requirements management is firstly defined in terms of its activities and also in terms of its results. This is the introduction to a discussion that is the main part of

this book. After the main discussion involving all the other disciplines, requirements management according to HOOD will be redefined in the more advanced, inclusive, and sophisticated way.

Part 2 starts with Chapter 6 Project Management. Project management is introduced, and its relationship to requirements development is investigated. The overlapping responsibilities between a project manager and a requirements manager are discussed, and the information common to both disciplines is exposed. The similarity and differences between tasks on a project plan and the requirements in a specification are explored.

Chapter 7 is Configuration Management. Configuration management is introduced and its relationship to requirements development is investigated. The overlapping responsibilities between the roles of configuration management and a requirements manager are discussed, and the information common to both disciplines is exposed. The idea of requirements and related information as configuration items and a set of requirements as a specification are explored.

Chapter 8 is Measurement and Analysis. Measurement and analysis is introduced, and its relationship to requirements development is investigated. The role of measurement and analysis is discussed, and the information common to measurement and analysis and requirements development is exposed. The similarity and differences between measurements of requirements and the requirements in a specification are explored. Particularly the need for measuring to support an aim rather than measuring just because a measurement is possible is emphasised.

Chapter 9 is Risk Management. Risk management is introduced, and its relationship to requirements development is investigated. The overlapping responsibilities between a risk manager and a requirements manager are discussed. The information common to both disciplines is exposed. The different types of risks and risk mitigation and their influence on the requirements in a specification are explored, as is the influence of requirements as a source of risk.

Chapter 10 is Test Management (Verification and Validation). Test management is introduced, and its relationship to requirements development is investigated. The overlapping responsibilities between a test manager and a requirements manager are discussed, and the information common to both disciplines is exposed. The similarity and differences between test cases in a test plan and the requirements in a

specification are explored. The difficulties encountered when using a test plan as a requirements specification are recounted.

Chapter 11 is Change Management. Change management is introduced, and its relationship to requirements development is investigated. The overlapping responsibilities between a change manager and a requirements manager are discussed, and the information common to both disciplines is exposed. The similarity and differences between change requests and the requirements in a specification are explored.

Chapter 12 is Advanced Requirements Management: the complete specification. This chapter is the summary of all previous chapters in Part 2, showing requirements management as a complete specification of the interfaces between requirements engineering and other systems engineering disciplines.

Part 3 is a practical guide, helping to motivate successful implementation of requirements driven improvements with HCM. Part 3 starts with Chapter 13 The HOOD capability models. This chapter provides motivation for supporting change. Psychological reasons why we need to measure and publish progress or lack of progress is described. Chapter 13 The HOOD capability models is an introduction to Chapters 14 and 15 which describe the HOOD capability models in detail.

Chapter 14 is HCM for Requirements Definition. The aim of this chapter is to introduce and define HCM for requirements definition. Help is offered for introducing improvements in practical terms. A stepwise introduction is recommended and supported, rather than present a theoretical treatise.

Chapter 15 is HCM for Requirements Management. The aim of this chapter is to introduce and define HCM for requirements management. Help is offered for introducing improvements in practical terms. A stepwise introduction is recommended and supported. This chapter pulls together the threads of all previous chapters and weaves the themes together to create a tapestry of all the ideas presented thus far. Each thread remains distinct but still takes its place in the overall picture.

1.5 How to read this book

There are many ways to use this book. Consider it a resource from which we may take as we please. The book may be read from beginning to end or

it may be used as a reference book and dipped into time and again with random access as a project progresses.

The book is divided into three parts. Part 1 is an introduction and sets the scene for the rest of the book. Part 2 deals with systems engineering disciplines and their relevance to requirements management. This is the main part of the book that you will use as a reference for technical information within your project. Part 3 deals with the challenge of introducing the ideas of Part 2 into your organisation. Part 3 is the part of this book that you will use as a reference for management information to support your team during the introduction of improvements.

Reading from beginning to end is particularly useful not only for those beginning to grapple with the complexities of requirements management. Also, those with many years experience might enjoy seeing things explained that previously had been taken for granted. You may agree with some or all of the various opinions presented. We hope that if you at first disagree with some of the opinions that this will help you to respect the fact that there are many dearly held opinions, and that our industry is still in its infancy and there is much still to be standardised. We have tried to represent various views while being sure to tell you what we have found to work in practice. In cases where there is disagreement we fall back on the author's years of experience in engineering since 1977 and use our common sense. Remember that this subject is no longer a technical challenge; we are dealing with peoples' understanding and people's failure to understand. We are dealing with peoples' weaknesses and insecurities. So please be kind to those with a different opinion to yourself. As my Father said, "You can learn something from any idiot". Your challenge at work is to include others and to strengthen the team. We hope this book helps you better to construct and explain your arguments in order that you can convince. We hope this book helps you better to understand the opinions of others that you can benefit from a broader view.

Using the same principal as is used in good training courses, information will be introduced in context to be defined later in detail. This organisation of information enables and supports the reading of the book from beginning to end. Learning is supported and encouraged by introducing topics in a broad way. By using a technique similar to active listening, the reader is encouraged to ask questions. By asking questions the answers to the questions become much more relevant than if the information is presented in a series of unrelated and surprising facts. By removing the element of surprise as another piece of information is presented learning is supported. In this book when information is dealt with in detail there is no surprise as we have prompted the questions by the manner of the introduction.

Consider your brain to be a series of cupboards, drawers, and shelves. The stuff you store on shelves is easily visible and can be found without much structuring. You just have to scan the shelves for what you want, and when you see what you are looking for your search is over. But your shelf space is limited, and anyway, even if you could replace all of your drawers and cupboards with shelves the search would take too long. Only having shelves for storing information would be like the advice given in the comic Viz, "If you have trouble finding things, just thread all of your possessions onto a piece of string. If you lose anything, just follow the string from one end to the other and you will be sure to find what you are looking for." Now, that is a technical challenge!

So we need more than just shelving. We need more structure. If we have lots of things put away we need to file things in some order so that we can find them again. We need cupboards so that we can store different things in different places. The cupboards may contain shelves but these are only visible when we have opened the cupboard door. If we do not later shut the door properly the contents may fall out.

The early chapters can be considered to be an explanation of which cupboards we have. By explaining the aims of each chapter and each part of each chapter we open the cupboard doors so that you may file the information away in your brain. By summarising each chapter we help you to close the cupboard door to prevent the contents falling out. The aims at the beginning and the summary at the end of each chapter support learning.

The modular form of this book supports the reading of the book as a reference book so that detail is easy to find just when your project needs it.

2 Why Requirements Management and Engineering

2.1 General

To ask why requirements management and engineering should be used is rather simplistic. A better question would be: "Why and when is it recommended and important to use RM&E methods and processes in projects?" This, however, is an excessively long title.

In recent years, management started to give RM&E more and more importance. For instance, management may require the introduction of requirements management in an organization by a specific deadline .

This approach implies that until that point there was no requirements management in the organization at all, but this is never true:

Every organization offering a product or a service practices requirements management. If this were not be true, the organization would have perished a long time ago. All organizations have a relationship between customer and contractor. And customers have aims that they are trying to achieve by using the contractor's products and services.

If there is the aim to fulfil the customer's wishes with products and services, requirements management takes place – at least implicitly.

Why is this topic more important for some organizations than for others? Why did the implementation of RM&E methods and processes gain so much importance during the last decades?

If we investigate small companies that develop a relatively simple product, requirements management is typically not a big deal. In such organizations, the development department usually communicates directly with the customer. They know the customer and their needs as well as the product or service to be developed.

Under these circumstances, it's just a small step from customer need to product or service, as only a small number of people is involved. The team members can communicate verbally and have little difficulty coordinating. The complexity of the product or service isn't high enough to require division of labour. Under these circumstances, acceptable and even good results can be achieved.

But if an organization grows rapidly, if the complexity of the products increases and if separation of labour becomes more common, the importance of using structured methods and processes in requirements management increases as well. Modern RM&E methods originate, not surprisingly, in the air and space flight industries. After all, these industries have always been pioneers in designing highly complex system that are highly coupled. On top of that, development is highly distributed which means that small coordination problems can produce fatal and expensive consequences.

In the last decades, the importance of electronics in vehicles increased significantly. The functionalities in electronics are a major distinguishing factor between car manufacturers. The method described earlier – to keep the requirements solely in the heads of the people – worked fairly well initially. But there is a limit, at which even the "local heroes" of an organization loose sight. This is even more fatal, as this process happens silently. But at some point, the number of errors becomes overwhelming and the costs explode.

To sum it up: The methods and processes of RM&E are indispensable for organisations that develop complex products or services using separation of labour.

There is one issue, however, that makes it very difficult to introduce RM&E methods into projects: A neglect of RM&E in the starting phase of a project materializes only during the final phase of a project.

In his book Software Engineering Economics, Prentice-Hall, published in 1981, Böhm states that the cost for fixing errors raise drastically, the later in the development process they are discovered. Today this is considered common knowledge.

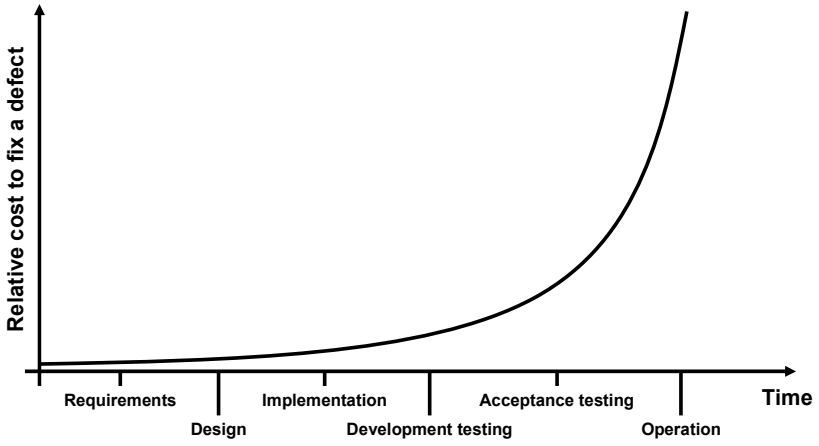


Figure 2.1: Relative cost of fixing a defect

This finding is even more significant, if we look at the areas where errors are found.

F. Sheldon analyzed a US-Air Force project, where 40% - 60% of all errors were found in the requirements. Only a third of the errors were found in design and code.

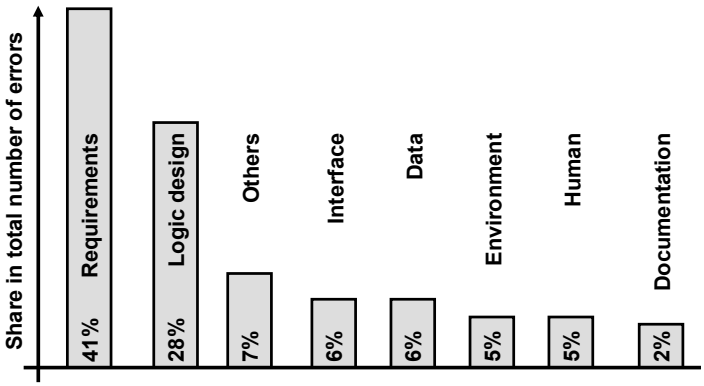


Figure 2.2: The share of errors in requirements in total number of errors

In other words: The highest savings can be achieved by focussing on finding errors – or avoiding them in the first place – during the early stage of a project by consequently using requirements management.

The advantages of RM&E methods and processes are even more visible if we investigate the connection to other disciplines, like project management, version management, configuration management, design, architecture, solution finding process, purchase, supplier management, customer service, distribution, marketing, as well as test management.

2.2 Advantages of RM&E in project management

2.2.1 Advantages of project planning

The aim of project management is to produce in the required quality with a calculated time and money budget. Typically, the estimate is documented in a contract.

A serious estimate can only be produced, if contractor and customer agree on the requirements of the system to be developed, at least roughly. Supposedly there are project managers or sales people who are capable of creating a time frame and cost estimate contractually, without even formulating the requirements of the system under development. But this

approach reduces the chances of success for the project from the very beginning.

Highest priority goal should be to develop as soon as possible a mutual understanding of the system under development between customer and contractor. This will serve as the foundation for a serious estimate of cost and time.

We should wonder why, in spite of this insight, countless offers contain cost and time estimates, but only a vague notion of the deliverable. It may have to do with the fact that collecting and eliciting the requirements for the first round of estimates takes a significant amount of work that neither the contractor nor the customer allocated a budget for. The start of the implementation is often the perceived starting time of a project. But this doesn't reflect reality: The moment when the decision of developing a new system had been made should be considered to be the starting point of the project.

According to RM&E best practice, 40% of the development time should be allocated for specifying. This value is based on experience. This reduces implementation time to 20%, leaving the remaining 40% for testing.

What does this mean for the creation of offers? The customer should allocate a budget of time and money before the bidding even starts, in order to produce a good requirements specification that will serve as a basis for the call for bids. Likewise, the contractors must have enough time to analyse the requirements specification and to write a target specification.

Both sides benefit from this approach, as a common understanding in regard to the system under development is being established. This will improve the quality of the project plan (time and budget), and ultimately for the implementation phase, which tends to be more expensive. The result is a more precise calculation of the system architectures, optimal resource planning, and avoidance of aberrations (based on misunderstandings). All this results in a reduction of costs

2.2.2 Advantages during the implementation phase

There is a high risk that changes in requirements won't be documented in the implementation phase. There are many reasons for this which are detailed in chapter 11 of this book.

In a proper realization of RM&E best practices, a process should be defined that regulates the handling of changes during all stages of the project.

Advantages for project management are manifold. For instance, contractor and customer both have an up to date project plan that they agree upon. All relevant stakeholders always have an up to date project status. Conflicts that are based on different understanding of the services to be rendered are thus avoided in the first place. Coordination of tasks is simplified, as changes are being discussed and decided upon in the open. All project tasks refer to a consolidated specification and can be traced to the implementation of the solution. Due to prioritizing of requirements, unforeseen influences can be dealt with quickly by adjusting the project plan. In other words, the methods of RM&E support the project manager's responsibilities.

2.2.3 Advantages during the acceptance phase

The final acceptance, also called buy-off, typically marks the end of a project. But acceptance can only happen if the acceptance criteria are fulfilled. The criteria, in return, should have been formulated together with the requirements.

If the specification has not been updated during the course of the project, it is difficult to define acceptance test, or to perform a final acceptance in the first place.

An up to date specification is the basis for test planning. Test cases can and should be developed based on the requirements. Feedback in the form of test results allows the project manager to estimate the actual quality of the product under development, and whether the project is on schedule.

A defect list based on tests is the foundation for making the decision of finishing a project.

There are real projects that continued on for years, because contractor and customer couldn't reach agreement on the final acceptance, caused by an outdated specification.

2.2.4 Advantages in regard to version and configuration management

Version and configuration management is a complex field in itself. Every business that continuously develops products consisting of components must deal with this topic.

Typically, there is a concept that regulates the versioning and configuration of products. There is the definition of a product structure, a component structure and an organization structure.

RM&E must develop an information infrastructure for specifying requirements based on these concepts, corresponding to the structures of organization, product and development.

Definitions for criteria must be defined that associate requirements with product versions, product variants, module versions and module variants.

The advantages are obvious:

- Product management is aware of dependencies of requirements and knows at every time which requirement is implemented in which product version or variant
- Product management can estimate the implications of changes on product versions or variants
- Key requirements are specified only once; changes are mapped automatically on the relevant development projects
- Problems with the respect to implementation or realization can be traced to specific product versions and variants
- Specific requirements can be reused for multiple product versions and variants

2.3 Advantages for finding solutions in design and architecture

2.3.1 Advantages design and architecture

Agile methods are a hot topic in software development. Here, a system with minimal basic functionality is developed at the beginning of the project. Subsequently, the system is extended in small steps in close collaboration with the customer. This stands in stark contrast to the waterfall model in software development, where the system is specified completely before implementation starts.

As so often, reality is in between these two approaches. For instance the fundamental general conditions and aims of the project shouldn't change. To exaggerate, a system shouldn't start as a bookkeeping system to become a CAD System during the course of the project.

On the other hand, changes must be allowed during the implementation phase.

Everybody who ever built a house knows what is meant by this. The fundamental architecture must stand before implementation starts (e.g. number of floors, location, type of heating, etc.). Nevertheless, some adjustments during the construction phase must be possible (e.g. the type of doors and windows, partitioning of rooms, etc.).

In other words, the basic requirements must be clearly defined and consolidated and be sufficient to build a stable architecture and design. Without this, an expensive redesign of the architecture may be necessary later on in the project. RM&E methods help through elicitation, modelling, analysis and review to create a foundation of these fundamental requirements. This will give the system a stable and sustainable architecture.

During development, RM&E helps to accommodate changes that will come without doubt. This is supported by structured processes. Working based on different states of the specification is a common problem especially in big teams. This discrepancy often shows up only during integration. RM&E methods help to execute change management in a structured fashion, which helps avoid erroneous development in its early stages.

2.3.2 Advantages in finding solutions

The key here is the capability to innovate. Especially companies and developers who used a specific technology for a long time for solving customer problems have the tendency to keep improving the technology in use, rather than to consider new avenues.

Examples help to explain this in detail. A central user requirement may be: “The user must be able to see the picture the most 5 minutes after taking it.” This user requirement could be realized thanks to the development of the Polaroid camera in the late 40s. It is based on the technology of special films that develop and fixate the photograph. The technology was improved over the decades, until digital cameras entered the market. Digital photography was a disruptive technology that solved the actual user requirements better and much cheaper than the technology based on photographic film. Bottom line: The Polaroid camera was almost completely replaced by digital cameras and is only left in a few niche markets.

There are numerous examples like this.

RM&E support the ability to innovate by providing elicitation methods for requirements on all levels and their traceability. To realize this, user requirements are specified in a way that they don’t contain any unnecessary limitations or solutions, due to technology or otherwise.

2.4 Advantages in purchase and supplier management

Typically, the purchasing department insists on a clear description of the scope of work for the system to be developed. This simplifies comparing offers from suppliers based on the work description, or user requirements specification. Ideally, the specifcation is sent to various suppliers, who in return create offers. The purchasing department can then pick the cheapest offer.

What are the advantages for the purchasing department? It can focus on the commercial aspects, like selecting the offer with the most favourable conditions and the cheapest price.

RM&E can only support the purchasing department with some aspects. For instance, it can support the creation of the user requirements specification, which is then sent to all potential suppliers.

But in addition, it is expected that the suppliers create a target specification, where they elaborate on how they intend to solve the problem. The department that commissioned the project should support the purchasing department in the evaluation of the target specification. Thus, the ultimate decision is made by the purchasing and commissioning departments together, and price is just one aspect in the decision making process.

The advantage of this approach is that the decision has a solid foundation, simplifying significantly the purchasing negotiations and supplier assessments.

2.5 Advantages in customer service, sales and marketing

Customer service can be an indicator of the service quality of an organization. Customer service typically has great interest in the high quality of the product delivered. If this is not the case, the effort of customer service will be unproportionally high compared to other departments.

Why is this? If problems are recognized by customers after delivery, customer service must take care of them. The most expensive problems are those where customer service must do this on site for a large number of users. But customer service has to deal with more than just product defects: They also deal with complains regarding lack of functionality or bad usability.

This makes customer service a universal communication channel that allows the company to gather feedback regarding the quality of the

product. Unfortunately, often this channel is underutilized for product improvements.

If customer service is used as the source for product requirements, the effort and cost of customer service could be lowered.

Customer service can also be used as a source of requirements for performing system diagnoses, which is another advantage of RM&E methods. Upon a customer service request, a potential system error must be identified quickly, so that it can be fixed without delay.

This, too, reduces effort and thereby costs for customer service.

Other beneficiaries of implemented RM&E methods are sales and marketing. Those are the groups that are typically the departments in an organization that are the closest to the customer. This makes it particularly important that requirements from the customers are not only captured precisely and accurately, but also that they get clearly communicated to the development department.

Communication is the most common problem in this area. Sales and marketing on one side and development on the other side need to have a solid foundation for talking to each other. It is often difficult to establish a common understanding regarding the system under development. But this is a crucial precondition for the development of a successful system. RM&E methods support the selection of appropriate modelling techniques and specification languages, which for the foundation for communication and avoid ill developments.

2.6 Advantages in test and verification management

The department for testing and verification management benefits the most from the active use of RM&E methods. The main task of this group is not only to asses the quality of already developed products and systems, but to accompany the whole development process and to verify intermediate results like specifications.

In order to execute this task, a reference between the current and desired state must be established. Thus, the definition of the desired state for the system under development is a prerequisite. Typically, this is captured with a specification. But far too often specifications are neglected and have a low quality. They tend to have gaps, are out of date or missing completely. RM&E not only accounts for verification of the system under development, but of the specifications as well, thereby creating a foundation for the creation of test cases.

Has this been missed, only few options are left: to develop new test cases through drawn-out interviews with the stakeholders or development department, or simply to use “common sense” and hope for the best.