

# PREFACE

---

While a commonly accepted concise definition of the term 'Requirements Engineering' is yet to be defined, it is widely agreed that

Requirements Engineering deals with activities which attempt to understand the exact needs of the users of the software system to be developed and to translate such needs into precise and unambiguous statements which will subsequently be used in the development of the system.

Requirements Engineering is becoming the key issue for the development of software systems that meet the expectations of their customers and users, are delivered on time and developed within budget.

Since the mid 1970's when Requirements Engineering was established as a distinct field of investigation and practice, a great deal of progress has been made in the methods, techniques and tools used within this important phase of software development. Despite this however, a significant gap exists in terms of theories and technology between on one hand the activities pertaining to the specification of requirements and on the other hand those activities concerned with the design and implementation of software systems. Today, requirements are still, in many cases, collected, analysed and translated to software thanks to excessive informal interaction between users and developers, trial and error, the ingenuity of a few individuals, often with failures which are more spectacular than the success stories!

In contrast to other areas of software development, research and practice in Requirements Engineering are fragmented. Although there is a vast literature covering individual facets of the

area, such as descriptions of tools, methods or techniques, each contribution falls into one of two categories: either it represents a prescriptive approach to requirements, normally as part of a development method of a wider scope than just requirements or it deals with a narrow set of issues from a particular philosophical or technological viewpoint. Furthermore, the coverage of the area tends to pay more attention to specification languages issues while, issues such as for example, understanding organisational aspects and their influence on software requirements or understanding requirements in terms of system properties, are virtually ignored.

This book is our response to these shortcomings. In our involvement in the field of Requirements Engineering as teachers, researchers and practitioners, we have very often, felt the need for reference material which would meet a number of objectives such as:

- providing a discussion of the issues, models, techniques and tools applicable to the field of Requirements Engineering within a general framework applicable to many different viewpoints
- covering both practical experience as well as research efforts in the area
- avoiding 'cookbook' solutions which so often describe the style adopted by many methods or tools

We believe that the book that emerged addresses all these objectives. Since the book appears in a 'Software Engineering' series, the discussion of the various topics in the book is influenced by the target of the Requirements Engineering in the form of software systems. An attempt however is made to relate requirements for such systems to the organisational and social settings within which they are intended to operate. The book is intended to serve the needs of different audiences in a balanced way by:

- supplying teachers and students of Information Systems/Software Engineering with material which can provide the basis for a stand-alone course on Requirements Engineering or as part of a more broad Systems Analysis/ Software Engineering course
- providing practitioners and researchers with state-of-the-art material on techniques, methods and tools for the elicitation, representation and validation of requirements.

Despite our research involvement in the field, we have opted for a rather detached stance on the issue called best practices in software Requirements Engineering. Without any intention to nominate the 'best' requirements engineering method, technique or tool, we attempted to make an uncompromising statement of facts based on the latest and most authoritative views on the subject as they appear in textbooks, journal publications, reports on standards and conference proceedings and which are of concern to the community of practitioners and researchers working in this area. We deliberately avoided including cookbook recommendations of requirements 'solutions' preferring instead an integrated treatment of the requirements *issues*, to avoid disorientating the reader in a maze of sometimes misleading, often contradicting, approaches. We believe that practitioners of such a complicated task as Requirements Engineering should first equip themselves with a thorough understanding of the best concepts and theories, then become exposed to a number of tools and techniques and finally formulate their own opinion about what works and what does not in the areas they practice. Also, for those who seek to advance Requirements Engineering beyond the current state-of-the-art, we hope that this book gives insights into the most important and fruitful paths to the unparalleled challenge posed by system requirements.

Establishing requirements for a software-intensive system involves two intellectual activities, analysis and specification. The former requires conceptual analysis of the needs of customer and user needs, their goals and assumptions whereas the latter is concerned with descriptions of the system behaviour and constraints placed on the system and its development by its environment. These activities are carried out in a social setting involving the requirements engineer, the builder of the system, the customer who commissions the system, the user who will eventually interact with the system and the personnel who will finally introduce the system in the enterprise.

The material in this book is presented from a system engineering perspective while recognising that the contextual setting of requirements engineering is a social one.

The book is organised around a framework which captures the pivotal aspects of Requirements Engineering, i.e. processes, models and tools.

Chapter 1 provides all the essential background and terminological knowledge required for the understanding of the material in succeeding chapters. Requirements Engineering is viewed from different perspectives, i.e. business, Software Engineering, and even from a cognitive perspective which describes the behaviour of requirements analysts.

Chapter 2 seeks to shed light into the confusion caused by different (and sometimes contradicting) terminology used for describing the same concepts within Requirements Engineering, by suggesting a framework for Requirements Engineering activities. The

framework views Requirements Engineering as a combination of three concurrent interacting processes which correspond to the three major concerns of eliciting knowledge related to a problem domain, ensuring the validity of such knowledge and specifying the problem in a formal way. The three succeeding chapters are devoted to these three topics.

In Chapter 3 the first of the Requirements Engineering processes, namely requirements elicitation is examined from the perspectives of concepts methods and tools. First the conceptual foundations of elicitation as a process in its own right are established, followed by a detailed discussion of approaches to elicitation, which range from traditional Systems Analysis techniques such as user interviews to the latest methods employed in disciplines such as Ethnomethodology and Knowledge Engineering.

Chapter 4 deals with another concern of Requirements Engineering, namely the development of conceptual models which specify the desired behaviour of the software system and the properties that the system must exhibit. Modelling principles and techniques are introduced which ensure that all the relevant information and concerns can be captured in a conceptual model. A requirements specification is viewed as a composite of three components: enterprise requirements, functional requirements and non-functional requirements. In this sense this chapter takes a wider, and in our opinion a more appropriate view of requirements specifications, than the traditional view of concentrating almost exclusively on functional requirements.

The three-fold view of the Requirements Engineering as elicitation, specification and validation is completed in chapter 5, where the process of requirements validation is covered. In a similar manner to preceding chapters, this chapter discusses the difficulties inherent in obtaining the users agreement on what constitutes a valid description of their problem, and presents methods, techniques and tools which attempt to overcome such difficulties. Following a discussion on the importance of validation within Requirements Engineering, this chapter introduces validation techniques such as prototyping, animation, and expert system approaches.

Chapter 6 focuses on the 'tools' aspect of the 'Concept-Method-Tool' view of Requirements Engineering. This chapter gives a historical overview of the role of Computer Aided Software Engineering (CASE) in Requirements Engineering. The multiple classifications of CASE technologies in this chapter aim to guide the reader into establishing criteria for selecting, integrating and using CASE tools for Requirements Engineering.

Requirements engineering is a discipline which addresses issues within both spheres of 'business' and 'software systems' and importantly it is concerned with the relationship

between the two. The need for rapid response to changing business environments, the employment of new approaches to organisational restructuring and the enabling influence of computers and communications lead us to believe that requirements engineering is an essential discipline of study and enquiry which brings systems engineering concerns closer to problems experienced in organisational settings. Requirements engineering is about addressing the problems associated with *business* goals, plans, process etc. and *systems* to be developed or to be evolved to achieve them.

In this book we have striven to cover a range of issues of importance to requirements analysis and specification in a non-prescriptive manner. To this end we have opted for a wide coverage of the subject concentrating on a discussion of the issues and current approaches to the problems being experienced in requirements engineering.

The book is aimed at students of undergraduate and postgraduate programmes with a substantial component of system development subject matter. The book assumes that the reader has already knowledge of system development techniques for either data-intensive or real-time systems.

## **Acknowledgements**

We wish to express our thanks to our colleagues who worked with us in exploring the exciting field of Requirements Engineering. In particular we wish to thank Janis Bubenko, Matthias Jarke, John Mylopoulos, Barbara Pernici, Colette Rolland, Arne Sølvsberg, Alistair Sutcliffe, Babis Theodoulidis with whom we had many informative and stimulating discussions on this topic.

We would also like to thank Vangelio Kavakli, Vana Konsta, Nikos Loucopoulos and Roger Smith for their help with case studies material. As always, Janet Houshmand has provided much appreciated administrative support. Finally, we would like to thank Rupert Knight and his colleagues at McGraw-Hill for their help and collaboration in the production of this book.

**Pericles Loucopoulos**

**Bill Karakostas**