THE ENGINEERING DESIGN OF SYSTEMS

THE ENGINEERING DESIGN OF SYSTEMS

MODELS AND METHODS

Second Edition

DENNIS M. BUEDE

WILEY

A JOHN WILEY & SONS, INC., PUBLICATION

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

Library of Congress Cataloging-in-Publication Data:

Buede, Dennis M.
The engineering design of systems: models and methods/Dennis M. Buede. – 2nd ed.
p. cm. – (Wiley series in systems engineering and management)
Includes bibliographical references and index.
ISBN 978-0-470-16402-0 (cloth)
1. Systems engineering. 2. Engineering design. 3. System design. I. Title.
TA168.B83 2009
620.001′171–dc22

2008022812

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

In memory of my Mother and Father

Contents

Preface *ix*

Part 1 Introduction, Overview, and Basic Knowledge *1*

Chapter 1 Introduction to Systems Engineering *3*
Chapter 2 Overview of the Systems Engineering Design Process *49*
Chapter 3 Modeling and SysML Modeling *73*
Chapter 4 Discrete Mathematics: Sets, Relations, and Functions *104*
Chapter 5 Graphs and Directed Graphs (Digraphs) *122*

Part 2 Design and Integration *149*

Chapter 6 Requirements and Defining the Design Problem *151*
Chapter 7 Functional Architecture Development *211*
Chapter 8 Physical Architecture Development *252*
Chapter 9 Allocated Architecture Development *284*
Chapter 10 Interface Design *319*
Chapter 11 Integration and Qualification *341*

Part 3 Supplemental Topics *373*

Chapter 12 Graphical Modeling Techniques *375*
Chapter 13 Decision Analysis for Design Trades *401*

Appendix A: Outline of Systems Engineering Documents *451*

Appendix B: IDEF0 Model of the Engineering of a System *455*

Glossary *475*

References *487*

Historical References *499*

Index *502*

Preface

This book is meant to be a basic text for courses in the engineering design of systems at both the upper division undergraduate and beginning graduate levels. The book is the product of many years of consulting on numerous portions of the system development process, research into the use of systems engineering in industry, and six years developing a course on the engineering design of systems. During the development of this book, I found that many engineers did not understand systems engineering. Even those that do may not have a good perspective on a complete and unified process for engineering a system. The desire to suppress the number of decisions being made during design is quite strong in most engineers. While engineers have learned modeling throughout their academic life, and most have developed models during the practice of engineering, very few engineers working on systems are knowledgeable of the modeling techniques required in systems engineering. In addition, most engineers are not aware of methods for using models during the systems engineering process. As a result, I adopted the following themes in formulating this book:

1. Defining the design problem in systems engineering is one of several keys to success and can be approached systematically using engineering techniques.
2. The design problem in systems engineering is defined in terms of requirements. These requirements evolve from a high-level set of mission and stakeholders' requirements to detailed sets of derived requirements.
3. The design process will fail if the requirements are defined too narrowly, leaving little if any room for design decisions and raising the possibility

that no feasible solution exists. The design problem should be well defined and decision rich.

4. For the design problem to be well defined, the evolving sets of requirements must be complete (none missing), consistent (no contradictions), correct (valid for an acceptable solution), and attainable (an acceptable solution exists). While it is not possible at this time to state requirements mathematically and prove these properties, it is possible to develop mathematical and heuristic representations of the design problem to assist in evaluating the presence of these properties.
5. The characteristics of the requirements will not be achieved if scenarios defining how the system will be used are not elaborated in detail, the interactions among the system and other systems are not defined, and the stakeholders' objectives are not understood. Each of these requires a different kind of modeling to be successful.
6. The design problem is not likely to be well defined if the requirements do not address every relevant phase of the system's life cycle.
7. The design problem is not likely to be well defined if the requirements do not contain stakeholder preferences for comparing feasible designs against each other.
8. The keys to understanding many of the modeling techniques for developing requirements, defining architectures, and deriving requirements are found in discrete mathematics: set theory, relations and functions, and graph theory.
9. Integration requires a well-defined design, including a design of the qualification process for verification, validation, and acceptance. A systematic process of design provides all of the necessary inputs for defining the qualification process.
10. Early validation of the evolution of the definition of the design problem needs to be pursued vigorously to ensure that the definition of the design problem does not change as the problem is defined in greater detail.
11. Qualification of the system is the key issue in integration. Qualification includes verification and validation of both the requirements and the system design, followed by the stakeholders' acceptance. There are many methods for qualifying the system; these methods must be chosen judiciously.
12. Successful qualification also requires that decisions about what should be tested be made in a systematic way that balances the two conflicting objectives of not wasting resources and obtaining stakeholder acceptance.

The major changes for the second edition are descriptions of The Object Management Group's Systems Modeling Language (OMG SysML™) and the introduction of new terminology. SysML is introduced in Chapter 1, defined in some detail in Chapter 3, and referenced in other chapters. The major changes in terminology were motivated by suggestions from readers to be less focused on specific application domains. Originating requirements has become stakeholders' requirements. Originating Requirements Document has become Stakeholders' Requirements Document. The operational architecture has become the allocated architecture. New material has been added in Chapter 1 to enhance the introduction of the engineering of systems. Additional material in Chapter 1 describes different types of systems and outlines the various attributes of the value provided by systems engineering. Minor changes have been made to several other chapters as well. Finally, I have added a large selection of historical references for systems engineering.

The book is divided into three major parts: (1) Introduction, Overview, and Basic Knowledge; (2) Design and Integration Topics; and (3) Supplemental Topics. The first part provides an introduction to the issues associated with the engineering of a system. Next, an overview of the engineering process is provided so that readers will have a context for the more detailed material. Finally, basic knowledge needed for the core material is presented. Homework problems are provided at the end of each chapter.

Chapter 1 defines a system, systems engineering, the life cycle of a system, and then introduces systems engineering processes. This material sets the stage for the details that follow.

Chapter 2 provides an overview of the details that are to come by presenting a number of basic concepts; these concepts include an operational concept, objectives, requirements, functions, items, components, interfaces verification, validation, and acceptance. The relations among these concepts are also addressed.

Chapter 3 provides an overview of modeling and the types of modeling needed in engineering systems. Modeling methods associated with SysML are then introduced and described. While IDEF0 is not part of SysML, this topic has been kept in Chapter 3 as an important part of the modeling concepts described in this book.

Chapter 4 presents basic discrete mathematics. The purpose of the discrete mathematics is to demonstrate the mathematical rigor for which systems engineering must strive and to provide a language with which we can discuss key issues. Examples of such important concepts are the distinction between a relation and a function and why this is critical for engineering a system; a partition of the elements of a set that can be applied to many systems engineering concepts (e.g., requirements); and partial orders of functional execution.

Chapter 5 extends the discussion of discrete mathematics to graph theory so that the graphical communication structures commonly used in the engineering of systems can be seen to have substantial problems as rigorous mathematical representations. On the other hand, the difficult concepts in Chapter 4 can be effectively represented with graphs for analysis and communication.

Part 2 covers the critical material required to understand the major elements needed in the engineering design of any system: requirements, architectures (functional, physical, and allocated), interfaces, and qualification.

Requirements development is approached as a systematic process in Chapter 6. This systematic process involves the definition of an operational concept of the system (including usage scenarios), a description of the involvement of the system with other systems, and an objectives hierarchy of the stakeholders across all phases of the system's life cycle. A partition of requirements is employed to discuss the systematic approach for defining requirements.

Definitions of the functional, physical, and allocated architectures are provided as well as the detailed methods for developing these architectures in Chapters 7 through 9. Chapter 7 begins with several definitions that are needed to enable a meaningful discussion of the topic. The notion of a functional architecture is defined. An emphasis is placed on process modeling in Chapter 7. However, additional material is presented in Chapters 3 and 12 on data and behavioral modeling methods, as well as other approaches for process modeling. (This material can be used while discussing Chapters 7 through 9.) Modeling approaches for partitioning a function into segments are discussed. Key topics are feedback and control within the functional decomposition and evaluating the architecture for shortfalls and overlaps. Chapter 7 also addresses the functionality needed for error detection and recovery as well as tracing the input/output requirements to functions and items.

Chapter 8 introduces the distinction between the generic and instantiated physical architectures. The morphological box is used to demonstrate the generation of multiple instantiated physical architectures. The graphical representation of the physical architecture is discussed along with notions of centralized, decentralized, and distributed architectures. Finally, fault-tolerant architectures are described.

Chapter 9 defines the allocated architecture and discusses the allocation of functions to components, the tracing and derivation of requirements, the analysis of activation and control structures, and the conduct of various analyses (risk, performance, and trade-off).

Chapter 10 characterizes interfaces; discusses the functions associated with interfaces in several contexts (communications systems and software design); describes interface architectures; and discusses interface design as it impacts system performance as part of the design process.

Finally, qualification of the system (Chapter 11) during integration requires the understanding of the stakeholders' needs and the qualification methods that are typically used. Deciding what to test and how to test it is critical in this phase of the development process. All of the topics in Chapters 6 to 11 are addressed in a rigorous and systematic manner, consistent with the general, practical application of systems engineering in industry.

Homework exercises are provided on each of these topics from Part 2 for several real but simple systems that are familiar to all students: an automatic teller machine (ATM), an air bag, and the OnStar system of Cadillac. A case study is available over the web to give the students a sample of the solutions to the homework. Readers are encouraged to access a limited version of a commercial system engineering software product (CORE) to enhance the conduct of these homework exercises and the educational mission of this book.

Finally, two additional key topics are introduced in the third part: methods for data, process, and behavior modeling and decision analysis. Chapter 12 addresses the topics of data modeling, process modeling, and behavior modeling. Many alternate approaches for each of these modeling areas are described in detail so that teachers using this text can substitute the material most relevant to their program for the IDEF0 process modeling in Chapter 3. (A few minutes of IDEF0 instruction will be required in any course because of the extensive use that I have made of an IDEF0 model of the systems engineering process in Appendix B.)

Chapter 13 presents the key topics of decision analysis as an integrative way of supporting the many decisions that are part of the design and integration of a system. These decision analytic topics include the development and quantification of values (objectives, value functions, and trade offs), and the modeling of uncertainty regarding facts.

The homework problems and the case study of the elevator are defined with the express purpose of having the student demonstrate the level of understanding necessary to perform the engineering activities described in the book. In developing these homework exercises I have taken the position that demonstrating an ability to discuss how to do systems engineering is a necessary but not a sufficient level of understanding. The CORE software (that is appropriate for use with this book is available via the web: http://www.vitechcorp.com) takes the tedium out of performing these systems engineering activities as well as reinforcing the basic concepts behind the activities. The case material related to an elevator system can be downloaded from the following web site: http://www.theengineeringdesignofsystems.com.

Many of the ideas for this book have originated with Alexander Levis. I have benefited greatly from my conversations with him. Jim Long introduced me to much of the systems engineering process and has since provided many thought-provoking concepts and ideas since we first met in 1991. Ron Howard guided me through the Ph.D. process and provided me with a wonderful foundation in decision analysis. This foundation in decision analysis is at the heart of the methods proposed in this hook. I have worked on several consulting over the last 20 years with Terry Bresnick; Terry's comments and questions have helped shape much of my thinking on the application of decision analysis to the engineering design of a system. Andrew Sage has seen several drafts of the book and provided many very useful comments, including suggestions for its title. Many faculty members who taught from the first edition have provided useful comments that led to improvements.

Sanford Friedenthal and Abe Meilich were kind enough to review portions of the original manuscript when it was near completion. Both Sandy and Abe provided very valuable comments for improving the quality of the material

and its presentation. Sandy has given me a great deal of information and encouragement to include the SysML material in this second edition.

Several colleagues at George Mason University and Stevens Institute of Technology have provided many useful comments and suggestions. I wish to thank Kathryn Laskey, William Miller, and Mike Pennotti.

Several students and teaching assistants have contributed to sections of these notes. Cathy Brown provided a substantial extension of the requirements for the elevator case study. John Van Ormer extended the physical architecture of the elevator. Jahan Araghi extended my initial case study on the ATM as part of his Master's project. Tong Zhang and Parham Pasha provided some examples on sets, relations, and graphs. Christine Salter provided extensive support in addressing topics that needed revision, developed solutions for homework problems, and provided solution material for the OnStar and ATM problems. Several student groups provided material on which the air bag case is based. Meg Giordana and Barry Liner provided extensive comments on the qualification material. Tim Parker developed two case studies for use in Chapters 8 and 9: the FBI Fingerprint Identification System and the Wide-Area Augmentation System of the Federal Aviation Administration. Steve Charbonneau provided interesting insights about state charts as part of his M.S. Thesis. The SYST 520 class at George Mason University during the spring of 1998 provided many extensive and useful comments on an early draft of the first edition.

I wish to thank all of these individuals, as well as many others with whom I have conversed on these topics, for stimulating me to complete this effort.

One of the most difficult aspects of writing this book has been to decide which material to include and which to leave out. There is still a great deal more to be said on the topics covered in this book and on some additional topics that were not included. More importantly, there is still a great deal more to discover, at least on my part.

Dennis M. Buede

Reston, Virginia
November 2008