A Practical Guide to SysML

The Systems Modeling Language

Morgan Kaufmann OMG Press

Morgan Kaufmann Publishers and the Object Management Group™ (OMG) have joined forces to publish a line of books addressing business and technical topics related to OMG's large suite of software standards.

OMG is an international, open membership, not-for-profit computer industry consortium that was founded in 1989. The OMG creates standards for software used in government and corporate environments to enable interoperability and to forge common development environments that encourage the adoption and evolution of new technology. OMG members and its board of directors consist of representatives from a majority of the organizations that shape enterprise and Internet computing today.

OMG's modeling standards, including the Unified Modeling Language™ (UML®) and Model Driven Architecture® (MDA), enable powerful visual design, execution and maintenance of software, and other processes—for example, IT Systems Modeling and Business Process Management. The middleware standards and profiles of the Object Management Group are based on the Common Object Request Broker Architecture® (CORBA) and support a wide variety of industries.

More information about OMG can be found at *http://www.omg.org/*.

Related Morgan Kaufmann OMG Press Titles

UML 2 Certification Guide: Fundamental and Intermediate Exams
Tim Weilkiens and Bernd Oestereich

Real-Life MDA: Solving Business Problems with Model Driven Architecture
Michael Guttman and John Parodi

Systems Engineering with SysML/UML: Modeling, Analysis, Design
Tim Weilkiens

A Practical Guide to SysML: The Systems Modeling Language
Sanford Friedenthal, Alan Moore, and Rick Steiner

Building the Agile Enterprise: With SOA, BPM and MBM
Fred Cummins

Business Modeling: A Practical Guide to Realizing Business Value
Dave Bridgeland and Ron Zahavi

Architecture Driven Modernization: A Series of Industry Case Studies
Bill Ulrich

A Practical Guide to SysML

The Systems Modeling Language

Sanford Friedenthal

Alan Moore

Rick Steiner

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Morgan Kaufmann Publishers is an imprint of Elsevier

Acquiring Editor: Rachel Roumeliotis
Development Editor: Robyn Day
Project Manager: A. B. McGee
Designer: Kristen Davis

Morgan Kaufmann is an imprint of Elsevier
225 Wyman Street, Waltham, MA 02451, USA

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods or professional practices, may become necessary. Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information or methods described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
Application submitted

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 978-0-12-385206-9

Printed in the United States of America

11 12 13 14 15 10 9 8 7 6 5 4 3 2 1

For information on all MK publications visit our website at www.mkp.com

Contents

Preface

Systems engineering is a multidisciplinary approach for developing solutions to complex engineering problems. The continuing increase in system complexity is demanding more rigorous and formalized systems engineering practices. In response to this demand, along with advancements in computer technology, the practice of systems engineering is undergoing a fundamental transition from a document-based approach to a model-based approach. In a model-based approach, the emphasis shifts from producing and controlling documentation about the system, to producing and controlling a coherent model of the system. Model-based systems engineering (MBSE) can help to manage complexity, while at the same time improve design quality and cycle time, improve communications among a diverse development team, and facilitate knowledge capture and design evolution.

A standardized and robust modeling language is considered a critical enabler for MBSE. The Systems Modeling Language (OMG SysML™) is one such general-purpose modeling language that supports the specification, design, analysis, and verification of systems that may include hardware, software, data, personnel, procedures, and facilities. SysML is a graphical modeling language with a semantic foundation for representing requirements, behavior, structure, and properties of the system and its components. It is intended to model systems from a broad range of industry domains such as aerospace, automotive, health care, and so on.

SysML is an extension of the Unified Modeling Language (UML), version 2, which has become the de facto standard software modeling language. Requirements were issued by the Object Management Group (OMG) in March 2003 to extend UML to support systems modeling. UML 2 was selected as the basis for SysML because it is a robust language that addresses many of the systems engineering needs, while enabling the systems engineering community to leverage the broad base of experience and tool vendors that support UML. This approach also facilitates the integration of systems and software modeling, which has become increasingly important for today's software-intensive systems.

The development of the language specification was a collaborative effort between members of the OMG, the International Council on Systems Engineering (INCOSE), and the AP233 Working Group of the International Standards Organization (ISO). Following three years of development, the OMG SysML specification was adopted by the OMG in May 2006 and the formal version 1.0 language specification was released in September 2007. Since that time, new versions of the language have been adopted by the OMG. This book is intended to reflect the SysML v1.3 specification, which was close to finalization at the time of this writing. It is expected that SysML will continue to evolve in its expressiveness, precision, usability, and interoperability through further revisions to the specification based on feedback from end users, tool vendors, and research activities. Information on the latest version of SysML, tool implementations of SysML, and related resources, are available on the official OMG SysML web site at http://www.omgsysml.org.

BOOK ORGANIZATION

This book provides the foundation for understanding and applying SysML to model systems as part of a model-based systems engineering approach. The book is organized into four parts: Introduction, Language Description, Modeling Examples, and Transitioning to Model-Based Systems Engineering.

Part I, Introduction, contains four chapters that provide an overview of systems engineering, a summary of key MBSE concepts, a chapter on getting started with SysML, and a sample problem to highlight the basic features of SysML. The systems engineering overview and MBSE concepts in Chapters 1 and 2 set the context for SysML, and Chapters 3 and 4 provide an introduction to SysML.

Part II, Language Description, provides the detailed description of the language. Chapter 5 provides an overview of the language architecture, and Chapters 6 through 14 describe key concepts related to model organization, blocks, parametrics, activities, interactions, states, use cases, requirements, and allocations, and Chapter 15 describes the language extension mechanisms to further customize the language. The ordering of the chapters and the concepts are not based on the ordering of activities in the systems engineering process, but are based on the dependencies between the language concepts. Each chapter builds the readers' understanding of the language concepts by introducing SysML constructs: their meaning, notation, and examples of how they are used. The example used to demonstrate the language throughout Part II is a security surveillance system. This example should be understandable to most readers and has sufficient complexity to demonstrate the language concepts.

Part III, Modeling Examples, includes two examples to illustrate how SysML can support different model-based methods. The first example in Chapter 16 applies to the design of a water distiller system. It uses a simplified version of a classic functional analysis and allocation method. The second example in Chapter 17 applies to the design of a residential security system. It uses a comprehensive object-oriented systems engineering method (OOSEM) and emphasizes how the language is used to address a wide variety of systems engineering concerns, including black-box versus white-box design, logical versus physical design, and the design of distributed systems. While these two methods are considered representative of how model-based systems engineering using SysML can be applied to model systems, SysML is intended to support a variety of other model-based systems engineering methods as well.

Part IV, Transitioning to Model-Based Systems Engineering, addresses how to transition MBSE with SysML into an organization. Chapter 18 describes how to integrate SysML into a systems development environment. It describes the different tool roles in a systems development environment, and the type of data that are exchanged between a SysML tool and other classes of tools. The chapter also describes some of the types of data exchange mechanisms and applications, and a discussion on the criteria for selecting a SysML modeling tool. Chapter 19 is the last chapter of the book, and describes how to deploy MBSE with SysML into an organization as part of an improvement process.

Questions are included at the end of each chapter to test readers' understanding of the material. The answers to the questions can be found on the following Web site at http://www.elsevierdirect.com/companions/9780123852069.

The Appendix contains the SysML notation tables. These tables provide a reference guide for SysML notation along with a cross reference to the applicable sections in Part II of the book where the language constructs are described in detail.

USES OF THIS BOOK

This book is a "practical guide" targeted at a broad spectrum of industry practitioners and students. It can serve as an introduction and reference for practitioners, as well as a text for courses in systems modeling and model-based systems engineering. In addition, because SysML reuses many UML

concepts, software engineers familiar with UML can use this information as a basis for understanding systems engineering concepts. Also, many systems engineering concepts come to light when using an expressive language, and as such, this book can be used to help teach systems engineering concepts. Finally, this book can serve as a primary reference to prepare for the OMG Certified System Modeling Professional (OCSMP) exam (refer to http://www.omg.org/ocsmp/).

HOW TO READ THIS BOOK

A first-time reader should pay close attention to the introductory chapters including Getting Started with SysML in Chapter 3, and the application of the basic feature set of SysML to the Automobile Example in Chapter 4. The introductory reader may also choose to do a cursory reading of the overview sections in Part II, and then review the simplified distiller example in Part III. A more advanced reader may choose to read the introductory chapters, do a more comprehensive review of Part II, and then review the residential security example in Part III. Part IV is of general interest to those interested in trying to introduce SysML and MBSE to their organization or project.

The following recommendations apply when using this book as a primary reference for a course in SysML and MBSE. An instructor may refer to the course on SysML that was prepared and delivered by The Johns Hopkins University Applied Physics Lab that is available for download at http://www.jhuapl.edu/ott/Technologies//Copyright/SysML.asp. This course provides an introduction to the basic features of SysML so that students can begin to apply the language to their projects. This course consists of eleven (11) modules that use this book as the basis for the course material. The course material for the language concepts is included in the download, but the course material for the tool instruction is not included. Using this course as an example course that introduces the language concepts, the instructor can create a course that includes both the language concepts and tool instruction on how to create and update the modeling artifacts using a selected tool. A shorter version of this course is also included on The Johns Hopkins site which has been used as a full day tutorial to provide an introductory short course on SysML. Refer to the End-User License Agreement included with the download instructions on The Johns Hopkins site for how this material can be used.

A second course on the same website summarizes the Object-Oriented Systems Engineering Method (OOSEM) that is the subject of Chapter 17 in Part III of this book. This provides an example of an MBSE method that can be tailored to meet the needs of specific applications.

An instructor may also require that the students review Chapters 1 and 2, and then study Chapter 3 on Getting Started with SysML. The student should also review the simplified MBSE method in Chapter 3, and create a system model of similar complexity to the Air Compressor example in the chapter. The student may want to review the tool section in the chapter to begin to familiarize themselves with a SysML modeling tool. The student should then study the automobile example in Chapter 4, and recreate some or all of the model in a modeling tool. Alternatively, if a modeling tool is not used, the students can use the Visio SysML template available for download on the OMG SysML website (http://www.omgsysml.org).

After working through this example, the instructor may choose to introduce one chapter from Part II during each following lecture to teach the language concepts in more depth. In an introductory course, the instructor may choose to focus on the SysML basic feature set, which is highlighted

throughout each chapter in Part II. The notation tables in the appendix can be used as a summary reference for the language syntax.

This second edition is also intended to be used to prepare for the OMG Certified Systems Modeling Professional (OCSMP) exams to become certified as a model user or model builder. The book can be used in a similar way as described above. For the first two levels of certification, the emphasis is on the basic SysML feature set. The automobile example in Chapter 4 covers most of the basic feature set of SysML, so this is an excellent place to start. In addition, each chapter in Part II shades the paragraphs that represent the basic feature set. In addition, the notation tables in the Appendix include shaded rows for the notational elements that are part of the SysML basic feature set. The unshaded rows constitute the remaining features that reflect the full feature set which is the covered in the third level of OCSMP certification.

CHANGES FROM PREVIOUS EDITION

This edition is intended to update the book content to be current with version 1.3 of the SysML specification, which was in the final stages of completion at the time of this writing. The changes for each SysML specification revision with change bars are available from the OMG website at http://www.omg.org/technology/documents/domain_spec_catalog.htm#OMGSysML. This update also includes marking of the basic feature set in Part II to differentiate it from the full feature set, and other changes to support preparation for the OCSMP exams. In addition, several other changes were made to this book to improve the quality and readability of the text and figures, and to incorporate additional relevant content. Some of the more significant changes are summarized below.

Chapter 3 is added in Part I and called Getting Started with SysML, to provide an introduction to a simplified variant of the language called SysML-Lite, as well as an introduction to a generic SysML modeling tool, and simplified MBSE method. The Automobile Example in Chapter 4 (previously Chapter 3) was revised to focus on the basic feature set of SysML, and is consistent with requirements for the OCSMP level 1 and 2 exams. Chapter 7 (previously Chapter 6) on blocks includes a significant rewrite to address the changes to ports and flows introduced in SysML v1.3. Chapter 9 (previously Chapter 8) on activities includes a new section on the Semantics of a Foundational Subset for Executable UML Models (fUML) which specifies execution semantics for activity diagrams. Chapter 18 (previously Chapter 17) on Integrating SysML into a Systems Development Environment, has been significantly rewritten to update existing sections and introduce new sections. The new sections include discussions on configuration management, auto-generation of documentation, a more elaborated discussion on transformations, and a summary of the SysML to Modelica Transformation specification. The modeling methods in Part III, include both the distiller example using functional analysis methods in Chapter 16 (previously Chapter 15) and the residential security example using the object-oriented systems engineering method (OOSEM) in Chapter 17 (previously Chapter 16). These chapters have been significantly refined to improve the conciseness and understandability of the methods and the quality of the figures.

Acknowledgments

The authors wish to acknowledge the many individuals and their supporting organizations who participated in the development of SysML and provided valuable insights throughout the language development process. The individuals are too numerous to mention here but are listed in the OMG SysML specification. The authors wish to especially thank the reviewers of this book for their valuable feedback; they include Conrad Bock, Roger Burkhart, Jeff Estefan, Doug Ferguson, Dr. Kathy Laskey, Dr. Leon McGinnis, Dr. Øystein Haugen, Dr. Chris Paredis, Dr. Russell Peak, and Bran Selic. The authors also wish to thank Joe Wolfrom and Ed Seidewitz, who contributed to the review of the second edition, and to Joe Wolfrom as the primary author of the Johns Hopkins University Applied Physics Lab course material on SysML and OOSEM referred to above.

SysML is implemented in many different tools. For this book, we selected certain tools for representing the examples but are not endorsing them over other tools. We do wish, however, to acknowledge some vendors for the use of their tools for both the first and second edition, including Enterprise Architect by Sparx Systems, No Magic by Magic Draw, and the Microsoft Visio SysML template provided by Pavel Hruby.

About the Authors

Sanford Friedenthal is an industry leader in model-based systems engineering (MBSE) and an independent consultant. Previously, as a Lockheed Martin Fellow, he led the corporate engineering effort to enable Model-Based Systems Development (MBSD) and other advanced practices across the company. In this capacity, he was responsible for developing and implementing strategies to institutionalize the practice of MBSD across the company, and provide direct model-based systems engineering support to multiple programs.

His experience includes the application of systems engineering throughout the system life cycle from conceptual design through development and production on a broad range of systems. He has also been a systems engineering department manager responsible for ensuring that systems engineering is implemented on programs. He has been a lead developer of advanced systems engineering processes and methods, including the Object-Oriented Systems Engineering Method (OOSEM). Sandy also was a leader of the industry team that developed SysML from its inception through its adoption by the OMG.

Mr. Friedenthal is well known within the systems engineering community for his role in leading the SysML effort and for his expertise in model-based systems engineering methods. He has been recognized as an International Council on Systems Engineering (INCOSE) Fellow for these contributions. He has given many presentations on these topics to a wide range of professional and academic audiences, both within and outside the US.

Alan Moore is an Architecture Modeling Specialist at The MathWorks and has extensive experience in the development of real-time and object-oriented methodologies and their application in a variety of problem domains. Previously at ARTiSAN Software Tools, he was responsible for the development and evolution of Real-time Perspective, ARTiSAN's process for real-time systems development. Alan has been a user and developer of modeling tools throughout his career, from early structured programming tools to UML-based modeling environments.

Mr. Moore is an active member of the Object Management Group and chaired both the finalization and revision task forces for the UML Profile for Schedulability and Performance and Time, and was a co-chair of the OMG's Real-time Analysis and Design Working Group. Alan also served as the language architect for the SysML Development Team.

Rick Steiner is an Engineering Fellow at Raytheon and a Raytheon Certified Architect. He has focused on pragmatic application of systems engineering modeling techniques since 1993 and has been an active participant in the International Council on Systems Engineering (INCOSE) model-based systems engineering activities.

He has been an internal advocate, consultant, and instructor of model-driven systems development within Raytheon. Rick has served as chief engineer, architect, and lead system modeler for several large-scale electronics programs, incorporating the practical application of the Object-Oriented Systems Engineering Method (OOSEM), and generation of Department of Defense Architecture Framework (DoDAF) artifacts from complex system models.

Mr. Steiner was a key contributor to the original requirements for SysML, the development of the SysML specification, and the SysML finalization and revision task forces. His main contribution to this specification has been in the area of allocations, sample problems, and requirements. He provided frequent tutorials and presentations on SysML and model-driven system development at INCOSE symposia and meetings, NDIA conferences, and internal to Raytheon.