Systems Engineering with SysML/UML

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

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

Systems Engineering with SysML/UML

Modeling, Analysis, Design

Tim Weilkiens

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann Publishers is an imprint of Elsevier

Acquisitions Editor: Tiffany Gasbarrini
Publisher: Denise E. M. Penrose
Publishing Services Manager: George Morrison
Project Manager: Mónica González de Mendoza
Assistant Editor: Matt Cater
Production Assistant: Lianne Hong
Cover Design: Dennis Schaefer
Cover Image: © Masterfile (Royalty-Free Division)

Morgan Kaufmann Publishers is an imprint of Eslsevier.
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

This book is printed on acid-free paper.

Library of Congress Cataloging-in-Publication Data
Weilkiens, Tim.
[Systems Engineering mit SysML/UML. English]
Systems engineering with SysML/UML: modeling, analysis, design/Tim Weilkiens.
p. cm. — (The OMG Press)
Includes bibliographical references and index.
ISBN 978-0-12-374274-2 (pbk. : alk. paper) 1. Systems engineering. 2. SysML (Computer science). 3. UML (Computer science). I. Title.

TA168.W434 2008
620.001′171—dc22
2007047004

ISBN: 978-0-12-374274-2

For information on all Morgan Kaufmann publications, visit our Web site at www.mkp.com or www.books.elsevier.com

08 09 10 11 12 13 10 9 8 7 6 5 4 3 2 1

Printed in the United States of America

Contents

Foreword by Richard M. Soley

According to the Boeing Commercial Aircraft Company, a Boeing 747-400 aircraft has a maximum gross take-off weight (including a typical 416 passengers, 171 cubic meters of freight in the cargo hold, and over 200,000 kg of fuel) of nearly 400,000 kg. Four behemoth engines push the bird at up to 88 percent of the speed of sound for unbelievable distances, up to 13,500 km, without refueling. The length of the aircraft alone (45 m) is longer than the Wright brothers' entire first flight.

But these amazing statistics, after 30 years finally to be eclipsed by even larger passenger aircraft, are nothing compared to the complexity of the system of systems which makes up the Boeing 747-400. The aircraft's electrical systems comprise some 274 km of wiring alone; high-strength aluminum and titanium parts are designed to work both standing still and heated by rapidly passing air. Backup systems keep navigation and life-sustaining systems running if primary systems fail; and even the in-flight entertainment system is one of the more complex systems on earth. In fact, the Boeing 747-400 comprises some 6 million parts, about half of which are simply fasteners like screws and bolts. It's no wonder that a Boeing spokesman once quipped, "We view a 777 as a collection of parts flying in close proximity." As a frequent flyer, I constantly and fervently hope that that proximity goal is respected!

All of these facts reflect a fact that all engineers understand: complexity is difficult to manage, and most interesting systems are complex. Worse, the compounding of systems into systems of systems—e.g., back to our airplane, the electrical, hydraulic, propulsion, lift surface, life support, navigation, and other systems of aircrafts—tends to introduce new complexities in the form of unexpected overlaps of parts and unexpected behavior of previously well-behaved systems.

The net result of the rising complexity of systems is the crying need for ways to express component systems in ways that those designs can be easily shared between design teams. A shared language is a crucial component of the design methodology of any system or process, be it electrical, software, mechanical, or chemical. And if that shared language is based on a pre-existing language, the team which must share design will be up to speed more quickly and more able to complete the design task—and the more-demanding long-term maintenance and integration task—better, faster, and cheaper.

This line of thinking was the thought process behind the Systems Modeling Language (SysML), an open standard developed and managed by the Object

Management Group (OMG) and its hundreds of worldwide member companies. OMG's SysML has quite a lot of positive aspects:

- It is based on (an extension of a subset of) the OMG's own Unified Modeling Language, or UML. Thus software developers comfortable with UML can move to SysML easily and tool developers with UML tools can easily support SysML as well.
- It is graphical. I have always been struck by the way software developers will share design with other developers using graphics (boxes and lines), but then write (for example) "C++ code" when communicating that design to a computer. Likewise product engineers will work out the details of a family of products using graphs, boxes, and lines but then outline those choice points using a textual approach, e.g. PDES/STEP. Graphical languages are a natural choice for design and have been used for thousands of years (witness building and ship blueprints, electrical circuit diagrams, etc.).
- It has many implementations. International standards simply aren't worth the paper they are printed on (or the hard disk space they occupy) if they are not implemented. Even before the SysML final specification became available in early 2006, several companies announced implementation of the standard.

It is important to understand that SysML is not just another software development modeling language. While software is an important component of nearly all complex systems today, it is almost never the only component—planes, trains, and automobiles all have software, and so do building controllers and chemical plants, but they also have plumbing, hydraulics, electrical, and other systems that must be harmonized and co-designed. Likewise the human and automated processes to use complex systems need to be mapped out as well in order to get good use from them (as well as maintain and integrate those systems for future requirements and into future systems). And all of these co-design issues need to take into account the fact that all complex systems are available in multiple configurations, based on facilities requirements, personnel hiring issues, and myriad other variables.

This is the focus of the SysML language, a language to model all of these different factors in the course of engineering complex systems and systems of systems. The purpose is not to mask the complexity of those systems, but rather to expose and control it in the face of constantly changing business requirements and constantly changing infrastructure.

SysML is likely not the last language for engineering complex systems; the fast-paced change in most engineering fields (and in fact introduction of completely new engineering fields, like bioengineering) will likely cause change in the specification language itself. This shouldn't be a surprise to any engineer; other long-lasting xispecification languages have changed over time. One simple example: building blueprints are several hundred years old, but those made before about 1880 didn't feature the international symbol for electrical outlets, since that feature didn't exist yet. Languages change to suit changing requirements in engineering, and

fortunately SysML itself is designed to change, by being based on a metamodeling infrastructure also standardized by the OMG, the Meta Object Facility (MOF). This same technology can be used to transform and integrate existing designs in older engineering modeling languages, including IDEF (a language heavily used in many fields, including aviation).

This brings us full circle, back to that collection of 6 million parts flying in close formation. There is no hope of developing such a large collection of parts—and people, facilities, hardware, software, hydraulics, navigation, and HVAC—without a shared design language which takes into account the constant change of component systems. SysML was specifically designed to address that need, and is already doing so in parts from tiny embedded robotics controllers to huge industrial components.

So enjoy learning the language. While not simple—complex problems often require complex solutions—it's logical and based on good engineering design practice. And keep those parts flying in close proximity!

Richard Mark Soley, Ph.D.
Chairman and Chief Executive Officer
Object Management Group, Inc.
Lexington, Massachusetts, USA

Author Biography

Tim Weilkiens works as a consultant and trainer for German consulting company oose Innovative Informatik GmbH. He is a member of the OMG working groups about SysML and UML and has written sections of the SysML specification. His son Ben helped energetic writing the book. You can reach both at twe@system-modeling.com