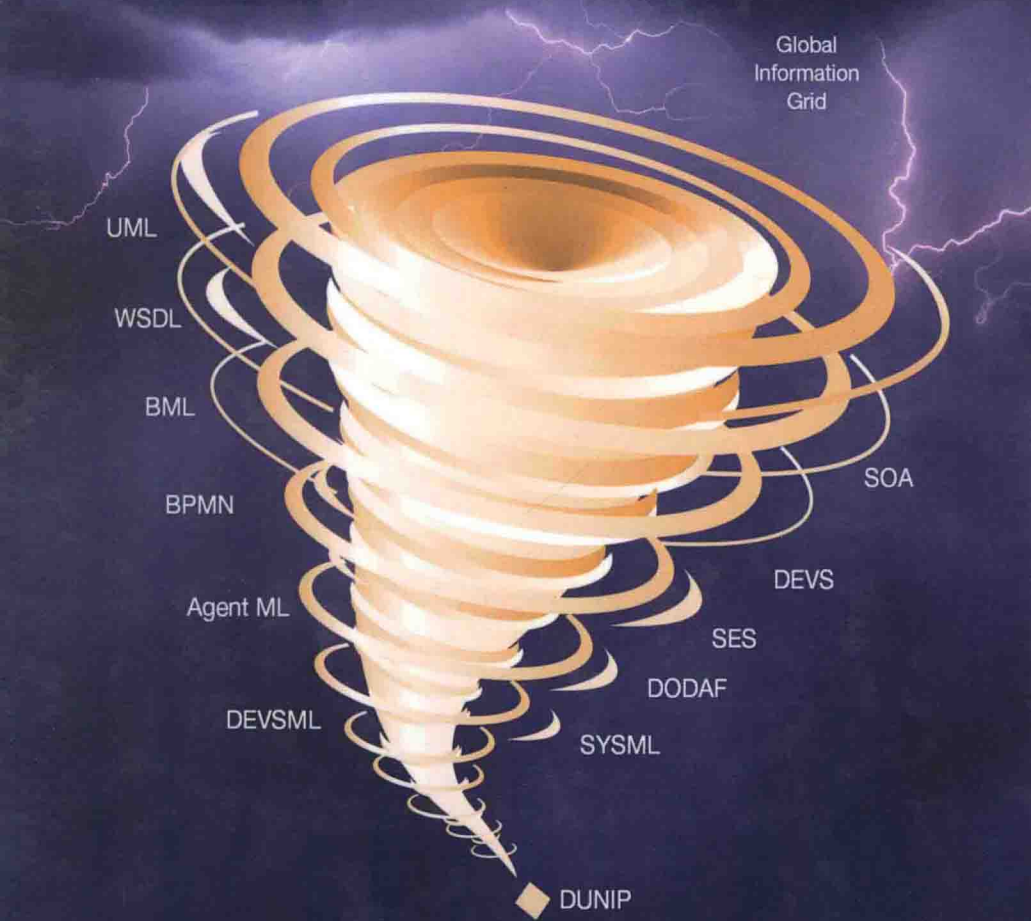
Netcentric System of Systems Engineering with DEVS Unified Process



Saurabh Mittal
José Luis Risco Martín

CRC Press
Taylor & Francis Group

Netcentric System of Systems Engineering with DEVS Unified Process

Saurabh Mittal
José Luis Risco Martín

With contributions by:
Deniz Cetinkaya • Alexander Verbraeck • Mamadou D. Seck

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **Informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Printed on acid-free paper
Version Date: 20121214

International Standard Book Number: 978-1-4398-2706-2 (Hardback)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Netcentric System of Systems Engineering with DEVS Unified Process

SYSTEM OF SYSTEMS ENGINEERING SERIES

Series Editor: Mo Jamshidi

Discrete-Time Inverse Optimal Control for Nonlinear Systems,
Edgar N. Sanchez and Fernando Ornelas-Tellez

Netcentric System of Systems Engineering with DEVS Unified Process,
Saurabh Mittal and José L. Risco Martín

Dedication

This book is dedicated to our families, who have never doubted our dreams, no matter how crazy they might be. Specially dedicated to our kids, who are at the age of asking the question "Why?" We love our kids as they are, and we love that question as scientists. The answer is usually reached in simulation,

... in our minds and ... beyond.

Preface

The foundation for this book was laid during Dr. Saurabh Mittal's doctoral study (2005 through 2007) at the Arizona Center for Integrative Modeling and Simulation (ACIMS), earlier known as the Artificial Intelligence Simulation Laboratory, in the Department of Electrical and Computer Engineering of the University of Arizona, Tucson, under the advisement of Professor Bernard P. Zeigler, the pioneer of Discrete Event Systems Specification (DEVS) formalism. Dr. Mittal was involved in multiple projects such as the modernization of the High Frequency Global Communications System (HFGCS) simulation model for the U.S. Air Force, the Automated Test-Case Generator (ATC-Gen) at the Joint Interoperability Test Command (JITC), Ft. Huachuca, and various other efforts. These projects were accomplished at ACIMS, which acted as a subcontractor to Northrop Grumman Information Technology (NGIT), JITC, Ft. Huachuca. The methodology adopted for all the projects had recurring themes of model specification, transformation, coding, testing, and execution. Most of the time, specifications directly led to executable code in DEVSJAVA (©). Sometime in 2006, Dr. Mittal coauthored an eXtensible Markup Language (XML) representation based on Finite Deterministic DEVS (XFD-DEVS) that allowed automated transformation of models into XML and executable code. Integration of XML with DEVS was a synergistic development as the ATC-Gen project also pushed the XML-based System Entity Structure (SES) ontology development. The research on platform-independent DEVS modeling was just beginning.

During the summer of 2006, Dr. José Luis Risco Martín arrived at the ACIMS Laboratory as a visiting research scholar. A strong collaboration was forged during his two-month stay in Tucson. Prior to coming at ACIMS lab, Dr. Martin had already developed the XML version of DEVS coupled models that acted as the common ground for the ensuing collaboration. Drs. Mittal and Risco Martín exchanged a lot of ideas, and the first version of the DEVS Modeling Language (DEVSML) stack was conceived at that point. They developed the execution of DEVS using Web Services technology and also developed the DEVS/Service Oriented Architecture (SOA) framework. They now had an architecture that takes in platform-independent XML-based DEVS model and executes them on a netcentric platform. The seeds of a platform-neutral DEVS Virtual Machine (DEVSVM) were laid.

During his doctoral study, Dr. Mittal developed many transformations for sources like XML-based state machines, structured natural language message-based systems, the Department of Defense Architecture Framework (DoDAF) (Version 1.0), Business Process Modeling Notation (BPMN), Unified Modeling Language (UML), XFD-DEVS, and so on, that all led to executable DEVS. Based on the bifurcated model-continuity-based life cycle methodology developed during the ATC-Gen project and various efforts in developing the DEVS back-end, the DEVS Unified Process (DUNIP) was conceptualized in 2007.

As a research scientist at L-3 Communications and a contractor to the U.S. Air Force Research Laboratory (AFRL), at Wright-Patterson Air Force Base, Ohio,

Dr. Mittal is now implementing the DEVSML stack at various AFRL initiatives. A new version, DEVSML 2.0, was proposed as a part of the Large-scale Cognitive Modeling (LSCM) initiative that integrates domain-specific languages (DSLs) with the prior version of DEVSML stack. A DEVS DSL called DEVSML, a platform-independent language in the DEVSML 2.0 stack, was also proposed that redefines the way DEVS models are written in a structured ecosystem such as Eclipse.

XML was not the only means to DEVS transformation, and Drs. Mittal and Risco Martín ventured into the domain of model transformations and model-to-DEVS transformation through a platform-independent DEVS DSL. Various tools were explored, most notably the Eclipse Modeling Framework (EMF), Eclipse Graphical Modeling Framework (GMF), Xtext, Xpand, Generic Modeling Environment, Groovy, Scala, Ruby, and so on that allow the development of DSLs. As DEVSML is an integral part of DUNIP, DUNIP was updated to include DSLs as a means to incorporate other domains that want to interface with a netcentric DEVSVM.

While Dr. Mittal was making advances at the modeling level, Dr. Risco Martín was making advances in the DEVS to non-DEVS interoperability using XML message communication and using the DEVS netcentric platform to run large-scale performance studies. Based on the advances and the capabilities of netcentric DEVS, their work was a strong force in the development of a DEVS Modeling and Simulation (M&S) Standard.

In 2011, Drs. Mittal and Risco Martín updated their server-side capabilities to the latest Web Service specification standards such as JAX-WS and developed advanced concepts for session replication and platform scalability. They also developed various domain-authoring tools in Eclipse and NetBeans ecosystems. This book is an outcome of the application of a lot of foundational ideas by the DEVS community and the latest in netcentric DEVS systems engineering by the authors. This book presents the latest on DUNIP and how it makes model-driven engineering (MDE) concepts applicable to the DEVS and at-large communities. DUNIP creates a possibility for wider adoption of DEVS systems engineering methodologies by integrating various DSLs, existing or non-existing, to a formal computational complex dynamical systems framework. This book is largely based on the authors' experience with the DEVS formalism and application of MDE in the context of netcentric system of systems.

HOW TO USE THIS BOOK

The authors expect the reader to be aware of the basic principles of object-oriented programming. The book is divided into five sections with each section covering a particular community of interest. A summary of each chapter is provided below.

Section I: Basics

Section I is designed for the undergraduate students and novices who want to venture into the DEVS world. The students can refer to some of the case studies in Section IV.

Chapter 1: Introduction to Systems Modeling and Simulation

Chapter 1 provides the basics of computational M&S. This chapter also provides an overview of UML and how various concepts of Object-oriented modeling are applied to M&S.

Chapter 2: System of Systems Modeling and Simulation with DEVS

Chapter 2 provides definitions of system of systems (SoS), complex systems, complex adaptive systems (CAS), and netcentric SoS. This chapter also provides the DEVS framework for M&S.

Chapter 3: DEVS Formalism and Variants

Chapter 3 describes DEVS formalism and its computational representation in Java. Various examples are provided that aid understanding of DEVS systems specification concepts.

Chapter 4: DEVS Software: Model and Simulator

Chapter 4 provides more examples and the computational representation of DEVS simulators. This chapter gives in-depth descriptions on how to write your own DEVS simulators. While this chapter and Chapter 3 give computational representations, Chapter 5 provides details on DEVS modeling without any programming language constructs.

Chapter 5: DEVS Modeling Language

Chapter 5 describes a DEVS DSL. This chapter is designed for early DEVS adopters who want to delve into the DEVS world at a much coarser level. This chapter provides an Eclipse editor to allow creation of platform-independent DEVS models. Various design issues of DEVS grammar are explained and its XML representation is also presented.

Chapter 6: DEVS Unified Process

Chapter 6 provides an overview of DUNIP and how it can be realized in a systems M&S effort. This chapter also describes the DEVSML 2.0 stack, which is the linchpin of netcentric DEVS systems engineering. This chapter binds the foundational concepts and presents a vision of an integrated spiral methodology using DUNIP.

Section II: Modeling and Simulation-Based Systems Engineering

Section II is for graduate students and advanced practitioners of DEVS. It is also for industry professionals who would like to learn the advanced capabilities of DEVS-based systems engineering methodology and to use MDE in their efforts. This section provides methodologies to apply M&S principles to SoS design and also provides an overview on the development of executable architectures based on architecture framework like DoDAF.

Chapter 7: Reconfigurable DEVS

Chapter 7 describes the variable structure capability in the DEVS component-based M&S framework. This chapter also presents the dynamic structure capability through the use of experimental frames and enhanced Model View Simulator Controller (MSVC) paradigm.

Chapter 8: Real-Time and Virtual-Time DEVS

Chapter 8 describes real-time and virtual-time DEVS and provides more details on the development of real-time systems.

Chapter 9: Model-Driven Engineering and Its Application to Modeling and Simulation

Chapter 9 is a collaborative effort with Deniz Cetinkaya, Dr. Alexander Verbraeck, and Dr. Mamadou D. Seck, all at the Delft University of Technology, The Netherlands. Deniz is a PhD student, Dr. Seck is an assistant professor, and Dr. Verbraek is a professor, and they have all contributed their expertise in sharing the authors' vision. This chapter formally presents a framework for model-driven development (MDD) as applicable to M&S domain.

Chapter 10: System Entity Structures and Contingency-Based Systems

Chapter 10 presents SES formalism, its computational representation, a constraint-based language, and a larger Knowledge-Based Contingency Driven Generative Systems (KCGS) framework for designing contingency-based artificial systems with DUNIP. The foundation of KCGS was laid at Cognitive Models and Agents Branch, AFRL, where a Cognitive Systems Specification Framework (CS2F) was developed, a realization of KCGS.

Chapter 11: Department of Defense Architecture Framework (Version 1.0)

Chapter 11 dives into the area of architecture frameworks and describes the U.S. DoDAF Version 1.0. This chapter analyzes the DoDAF specification document, finds gaps in the specification, and suggests augmenting the DoDAF 1.0 specification with two new documents that allow the M&S community to build executable architectures.

Chapter 12: Modeling and Simulation-Based Testing and DoDAF Compliance

Chapter 12 gives an overview on various model-based testing methodologies at various levels of system specifications and on how architecture frameworks such as DoDAF can be subjected to test and evaluation studies.

Section III: Netcentric System of Systems

Section III is for graduate students, advanced users of DEVS, and industry professionals who are interested in building DEVS virtual machines and netcentric SoS. This section can be read directly after Section I.

Chapter 13: DEVS Standard

Chapter 13 gives an overview of the complexities in standardization efforts at the modeling level as well as at the simulation level. This chapter also describes the integration of DEVS and non-DEVS (MATLAB®) systems. Examples are presented in sufficient detail.

Chapter 14: Architecture for DEVS/SOA

Chapter 14 describes architecture for DEVS/SOA and the DEVS Virtual Machine. This chapter provides details about the server and client designs and how the DEVSML 2.0 stack is implemented in a netcentric infrastructure.

Chapter 15: Model and Simulator Deployment in a Netcentric Environment

After an understanding of the netcentric DEVS is given in Chapter 14, Chapter 15 provides practical examples on how to deploy the models at the client side and the simulators at the server side. Various netcentric architectural constructs are described in sufficient detail.

Chapter 16: Netcentric System of Systems with DEVS-Based Event-Driven Architectures

Chapter 16 integrates the emerging paradigm of Event Driven Architecture (EDA) with the netcentric DEVSML 2.0, the DEVSVM, and MDE toward the development of agile netcentric DEVS-based EDAs. This chapter also provides details about integrating the Web Service Description Language (WSDL), a DSL, into the DEVSML 2.0 stack.

Chapter 17: Metamodeling in Department of Defense Architecture Framework (Version 2.0) Metamodel

Chapter 17 revisits the DoDAF with its latest version, which is oriented toward data engineering, sharing, and interoperability. This chapter represents DoDAF 2.0 viewpoints with the SES ontology and presents an argument that DoDAF 2.0 is now in a state where it can effectively act as a DSL that can be anchored to a formal M&S framework. This chapter addresses issues of interoperability and integration of DoDAF DSL based on SES ontology to the netcentric DEVSVM.

Section IV: Case Studies

Section IV presents the case studies that realize many of the concepts defined in Sections I–III.

Chapter 18: Joint Close Air Support: Building from Informal Scenarios

Chapter 18 provides a complete example using DEVSML. Joint Close Air Support (JCAS) is a U.S. Joint Measure Thread (JMT) that describes the JCAS scenario. The user is led from informal natural language specification through a requirements refinement process into a DEVSML specification that is executable. This chapter can be read in conjunction with Chapter 5.

Chapter 19: DEVS Simulation Framework for Multiple Unmanned Aerial Vehicles in Realistic Scenarios

Chapter 19 brings a realistic context in the specification of scenarios. This chapter presents a simulation framework executable on DEVS/SOA in the context of the modeling of multiple unmanned aerial vehicles. Instead of DEVSML, the model is

directly built into the implementation programming language. Various performance studies are conducted in a netcentric DEVS/SOA infrastructure. This chapter can be read in conjunction with Chapters 4 and 15.

Chapter 20: Generic Network Systems Capable of Planned Expansion: From Monolithic to Netcentric Systems

Chapter 20 presents an overview of the state-of-the-art HFGCS simulation model, also known as the Generic Network System Capable of Planned Expansion (GENETSCOPE). A methodology is presented that made a monolithic system, a modular system that is executable in a DEVS component-based framework. Further ideas are presented to make it netcentric. This chapter can be read in conjunction with Chapters 7 and 16.

Chapter 21: Executable UML

Chapter 21 takes a hard look at the UML framework and develops various transformations that take UML elements to the DEVS world and vice versa. This chapter can be read in conjunction with Chapters 5, 6, and 10.

Chapter 22: BPMN to DEVS: Application of MDD4MS Framework in Discrete Event Simulation

Chapter 22 takes BPMN to the DEVS executable using the MDD4MS framework developed in collaboration with Centikaya, Dr. Verbraeck, and Dr. Seck. This chapter describes a BPMN metamodel and the transformations to a DEVS metamodel. The metamodel is also integrated with the DEVSML metamodel. This chapter can be read in conjunction with Chapters 5 and 9.

Section V: Next Steps

Section V presents the application of DEVS extensions in a netcentric environment to the discipline of CAS.

Chapter 23: Netcentric Complex Adaptive Systems

Chapter 23 provides an overview of CAS and how modeling of CAS using DEVS concepts can be attempted. This chapter also conceptualizes a netcentric CAS and argues that a netcentric SoS becomes a CAS. Finally, this chapter provides various features and requirements to achieve integration and interoperability in a netcentric CAS.

Online

Various other supporting materials are available online at:
http://www.duniptechnologies.com/book/sos.

Exercises

Some of the chapters come with exercises that advance the idea and concept described earlier. A more serious student is encouraged to attempt these. Partial solutions to the exercises are available online at:
http://www.duniptechnologies.com/book/sos/exercises.

Software

Instructions to use and install the software used in this book are available online at: http://www.duniptechnologies.com/book/sos/software.

WHAT THIS BOOK IS NOT

This book is not about learning a programming language or a specific tool. It is about the application of MDE concepts and how to build models of complex dynamical that are executable on a DEVS platform. Specifically, this book will not help if you are looking to learn the following:

- Advanced programming skills in Java or associated implementation languages
- DSL development tools like Xtext, Xpand, EMF, Eclipse GMF, or eCore
- Server administration (Glassfish 3.x or Tomcat 6.x)
- Graphical user interface development

QUESTIONS AND CONTACT INFORMATION

For any questions, concerns, or errors in this book, please send an e-mail to: sosbook@duniptechnologies.com.

Acknowledgments

In this arduous journey of two years, we have to express many thanks to our families and our young kids. Both of us became parents during the manuscript preparation phase. We especially thank our wives for those moments in which they took care of the home and especially the kids, who are now toddlers, while we stayed away working on the book. We wholeheartedly thank our families for their unconditional love, patience, kindness, understanding, and encouragement.

We are grateful toward our series editor, Professor Mo Jamshidi, and the publisher, Taylor & Francis Group/CRC Press, for finding merit in our work and giving us an opportunity to develop it further.

We sincerely thank Professor Bernard Zeigler for his support in conceptualizing the ideas in the early stages of our work at the ACIMS Laboratory. We came to know many people at NGIT who helped create the foundation of DUNIP. We thank all our friends and colleagues at NGIT for their support.

We would like to thank Deniz Cetinkaya, Alexander Verbraeck, and Mamadou Seck at Delft University of Technology for collaborating on two chapters for this book. Their important contributions are very much appreciated.

We especially thank our friends and colleagues at L-3 Communications, AFRL, and the Department of Computer Architecture and Automation (Universidad Complutense de Madrid [UCM], Spain).

Finally, we thank our friends who collaborated on the development of certain ideas and experiments given in this book. We thank Jesús Manuel de la Cruz and Eva Besada (UCM), and Joaquín Aranda and Alejandro Moreno (Universidad Nacional de Educación a Distancia, Spain).

Saurabh Mittal
José Luis Risco Martín

Authors

Saurabh Mittal is the founder and principal scientist at Dunip Technologies, which he manages in his free time. He is currently a full-time research scientist at L-3 Communications and is a contractor to the U.S. Air Force Research Laboratory, Wright-Patterson Air Force Base, Ohio. Here in this capacity, he is working on large-scale cognitive M&S using DEVS systems engineering, cognitive domain ontologies extending SES theory and various other cross-directorate M&S integration and interoperability efforts using architecture frameworks such as Department of Defense Architecture Framework. He is a recipient of the highest civilian contractor recognition "Golden Eagle" award by Joint Interoperability Test Command, Defense Information Systems Agency, U.S. Department of Defense. He serves on various conference program committees and is a reviewer for many prestigious international journals. His areas of interest include web-based M&S using SOA, executable architectures, netcentricity, DUNIP, domain ontologies, platform independent modeling, domain specific languages, SES theory and applications, parallel-distributed simulation, complex systems, complex adaptive systems, and system of systems engineering using DoDAF.

José L. Risco Martín is an associate professor at the Computer Architecture and Automation Department of UCM, Spain. His research interests focus on the design methodologies for integrated systems and high-performance embedded systems, including new modeling frameworks to explore thermal management techniques for multiprocessor system-on-chip, novel architectures for logic and memories in forthcoming nanoscale electronics, dynamic memory management and memory hierarchy optimizations for embedded systems, networks-on-chip interconnection design, and low-power design of embedded systems. He is also interested in theory of M&S, with an emphasis on DEVS, and the application of bioinspired optimization techniques in computer-aided design problems.

Contents