

Preface

The primary goal of this book is to introduce students to the basic principles of object-oriented modeling, design and implementation of simulation models. Emphasis is on model implementation by reinforcing OO modeling and programming. The treatment of statistics is left to a minimum.

For undergraduate students (of science and engineering), another goal is to provide relevant material for easy understanding of object orientation and simulation as early as possible in their curricula.

To accomplish the goals mentioned previously, the book consists of three parts. Part 1 presents a basic introduction to general modeling and simulation development, then introduces modern commercial graphical simulation and animation software: Arena and Flexsim. Other commercial simulation software, such as Extend and ProModel are mentioned but are not explained in the book.

The second part presents an overview of Object Oriented modeling and programming and introduces OOP with a high-level programming and simulation language, OOSimL. In this part, the book basically explains object-oriented modeling and techniques for designing and implementing problem solutions.

This part of the book presents a review of the basic principles and techniques of OO programming, and helps to clarify most of the important object oriented concepts taught in programming courses. The problem solving principles and techniques are illustrated by introducing problem modeling and their solutions, which are implemented in OOSimL. The OOSimL language is based on Java and the design of its simulation facilities were influenced by the Simula language and DEMOS classes.

The third part of the book, presents the principles of discrete-event simulation with an object-oriented modeling and programming focus using the OOSimL language. The process style of simulation is emphasized. All the basic OO programming principles explained in Part 2 are applied to the simulation concepts and techniques.

This book takes the “objects early” approach; it presents the object-oriented principles from the beginning, and provides the reader with a stronger foundation with objects and classes. A few models are described with the Unified Modeling Lan-

guage (UML), other models are described with DEMOS diagrams and extensions developed by Pooley and Hughes.

The basic language constructs of OOSimL are presented gradually with the various programming principles and in an incremental manner. Most conventional textbooks on programming stress too much syntax of the programming language, e.g., Java, from the beginning. This results in unnecessary difficulty for the student in understanding the underlying concepts in problem solution and programming.

Standard pseudo-code constructs are explained and applied to various case studies. These are coded with the pseudo-code-like and high-level notation of OOSimL. For some examples with OOSimL, the Java implementation follows the explanation. General descriptions of the data structures needed in problem solutions are also discussed. All the object-oriented modeling and programming concepts are also applicable to the Java programming language.

The most recent version of the OOSimL simulation software, and the set of simulation models discussed in this book are available from the Web site:

<http://science.kennesaw.edu/~jgarrido/oosiml.html>.

There are very few books on object oriented simulation. This book and its associated software and models, were originally designed as a teaching tool. Professional practitioners can use the book to clarify and review the important practical concepts of performance modeling with simulation and some advanced application of programming in OOSimL and Java.

The book does not present the detailed theory of statistical treatment that appears in standard textbooks on system simulation. It is not a complete reference on performance measures of systems. The book only includes the necessary basic theory of probability to support the understanding of applying the appropriate probability distribution in the construction of the simulation models. The book also includes basic material that briefly presents the concepts and techniques of verification and validation of simulation models. Most other books on conventional discrete-event simulation include a more complete treatment on simulation theory, statistical analysis, validation, and on performance issues.

This book is aimed at college students in: computer science, mathematics, science, engineering, and management science.

The material presented in the book can also be used as a textbook for an applied course in object-oriented programming. More appropriately, this book can be used as a supplemental book for courses in advanced performance modeling and analysis. The main features of the book are the following:

- The basic principles of performance measures are applied to several types of systems using object-oriented modeling and the process interaction approach to simulation. Performance and Functional models are introduced.
- The process interaction approach to modeling and simulation are emphasized.
- The understanding of large and complex systems is facilitated.
- The understanding of modeling complex concepts and problems is simplified. This is applicable in modeling systems dealing with concurrency and timing constraints.

- The practical use of object-oriented concepts to solve real problems.
- The use of the simulation software as part of a larger integrated software development effort.
- An introduction to understand the complexities of modeling and simulation with OOSimL and Java. This may be convenient before deciding to acquire a more comprehensive and expensive simulation software system.

While teaching in the Department of Computer Science and Information Systems of Kennesaw State University, my colleagues and I have acquired experience in teaching object oriented programming at various levels. Discrete-event simulation has been taught, up until now, in an elective course at the senior level. Some of the difficulties encountered are the following:

- The teaching approach, which includes inappropriate material content for teaching students with no previous programming knowledge, the sequence of topics is not necessarily adequate, lack of basic software engineering principles in the course contents, and other factors.
- The textbooks on programming, which are mainly language-syntax oriented.
- The programming languages, most of which is mainly C++ or Java, are difficult to learn.
- The textbooks on simulation, most of which are too mathematical-oriented based mainly on statistical analysis.
- The development environment and other tools, which are difficult to learn and/or were designed for an industrial setting.

This book addresses part of these difficulties by providing material for improving the overall approach in teaching programming principles and adopting an “early introduction” approach to discrete-event simulation.

Problem solving and programming principles are emphasized from the beginning with pseudo-code. Pseudo-code is used for most program design. The OOSimL compiler is used to convert a program written in OOSimL into a Java program.

The construction of complete programs is carried out in the high-level language, OOSimL. This language has several advantages, it:

- Is higher-level and much more easy to read and maintain than Java or C++
- Enforces good structuring and documenting of programs
- Includes standard object-oriented constructs that are easy to apply
- Includes standard design structures for algorithm design
- Allows abstraction
- Is compatible with Java and provide a smooth transition to Java.
- It is used to implemented object-oriented discrete-event simulation models

Another goal of the book is to provide programmers a smooth transition to object-oriented modeling and a gradual introduction to object-oriented simulation.

For every topic discussed, one or more complete case studies are presented and explained with the corresponding case study implemented in the OOSimL (and Java) programming languages.

The book includes summaries, examples, and problems in every chapter. The case studies presented as simulation models, the simulation software, and the corresponding PowerPoint slides, will be available on the book Web page.

Acknowledgements

In developing the software, simulation models, case studies, and writing this book, I have had support from my NSF CPATH grant, which I share with Dr. Pamila Dembla.

I have received direct support from the Chair of the Department of Computer Science and Information Systems (CSIS), Dr. Donald Amoroso. I acknowledge direct and indirect help from faculty of CSIS through discussions on programming and various applications of simulation with several of my colleagues: Ben Setzer, Richard Gayler, Ken Hoganson, Chong-Wei Xu, Richard Schlesinger, among others.

Two of my students, Ernesto DiMarco and Ernesto Tham, initially investigated and developed the procedure and tests to configure the Eclipse environment for use with the OOSimL simulation language.

Marietta, Georgia
February 2009

José M. Garrido



<http://www.springer.com/978-1-4419-0515-4>

Object Oriented Simulation
A Modeling and Programming Perspective
Garrido, J.M.
2009, XIX, 431 p., Hardcover
ISBN: 978-1-4419-0515-4