Simulation Foundations, Methods and Applications

More information about this series at http://www.springer.com/series/10128

Okan Topçu · Halit Oğuztüzün

Guide to Distributed Simulation with HLA

Springer

Okan Topçu
Middle East Technical University,
Northern Cyprus Campus (METU NCC)
Kalkanlı, Güzelyurt, Mersin 10
Turkey

Halit Oğuztüzün
Middle East Technical University
Ankara
Turkey

ISSN 2195-2817 ISSN 2195-2825 (electronic)
Simulation Foundations, Methods and Applications
ISBN 978-3-319-61266-9 ISBN 978-3-319-61267-6 (eBook)
DOI 10.1007/978-3-319-61267-6

Library of Congress Control Number: 2017943250

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

To Tuğçe—the love of my life
and
To Oğuz—the meaning of my life

Okan Topçu

To Serpil, Çerağ and Ozan
My circle of love

Halit Oğuztüzün

Foreword

I have always enjoyed driving boats, but this was a very special day. For the first time in my life, I was driving a huge ship, the 203 meter long, 2500 passenger cruise ship Silja Symphony. I was about to enter the Helsinki harbor when I noticed a small vessel crossing my ship's path. I turned sharply starboard, but nothing happened. I kept turning, and after five seconds the ship reacted and steered sharply toward a quay, completely out of control. In just a few seconds, the ship would crash, causing a major disaster.

Fortunately, all of this took place in a simulator in a ship's pilot school. My heart pounded while I drove the ship, immersed in the virtual environment. A few minutes later, I calmed down over a cup of coffee while the instructor walked through my decisions and mistakes.

Simulations let us experience dangerous situations, and let us learn how to handle them without risking our lives. Training in simulators saves time and money, instead of using real ships or aircraft. Simulators let us try many what-if scenarios, analyze the results, and take better decisions. Simulations take us places where we otherwise cannot go.

Simulation programs were initially developed as monolithic applications by an individual or a group. As simulation technology advanced, several simulation applications could be interconnected. Expertise from several groups can be combined, making it possible to build even more powerful, scalable, and interactive simulations. More complex scenarios can be simulated, scenarios that are more like the real world. Physical equipment, human decision making, sensors, environment, communications, and more can be integrated into one holistic approach, solving important problems in today's complex world.

For more than twenty years, I have been involved in developing distributed simulation systems as well as supporting the evolution of standards for simulation interconnectivity. I have visited a large number of countries, giving advice to distributed simulation projects in places such as Japan, Brazil, Canada, Taiwan, China, USA, and most European countries. Everywhere I've been, I've seen a big demand for know-how about distributed simulation.

It is an honor for me to introduce Drs. Topçu and Oğuztüzün's book, which provides an introduction to simulation as well as valuable insights into the design of distributed simulation systems. This book starts with a history and overview of the

High-Level Architecture (HLA), the leading modern standard for distributed simulation. HLA provides services for information exchange and synchronization between simulations that together form a federation.

To successfully apply HLA to a specific domain, be it pilot training, space mission planning, transportation, or manufacturing, a Federation Object Model, or FOM, is needed. The FOM describes the information that the simulations exchange. The second part of this book covers this interesting area. Today, there are many FOM standardization efforts ongoing around the world, for example, the Space Reference FOM, air traffic management FOMs, the Real-time Platform Reference FOM, road traffic simulation FOMs, and the NATO Education and Training FOM. This part of the book gives a foundation for anyone who wants to understand these FOMs and the architecture of such distributed simulation systems.

In part three, Drs. Topçu and Oğuztüzün show how applications, known as federates, use HLA and how they are structured and developed. While the HLA specification describes each service, this book shows how to put everything together. Various tools and frameworks facilitate HLA development. Drs. Topçu and Oğuztüzün introduce one such framework.

After covering some advanced topics in part four, Drs. Topçu and Oğuztüzün conclude this book with a complete case study. This should be a good time for the reader to gain an understanding of how everything fits together.

Drs. Topçu and Oğuztüzün's book will be very valuable to a large number of simulation developers, ranging from students to advanced practitioners. This book also fills a hole, since an up-to-date book on HLA has been missing for several years. I strongly recommend this book and hope that the reader will enjoy it as much as I did!

2017

Björn Möller
Vice President and Co-founder of Pitch Technologies,
Vice Chair of the IEEE 1516 HLA standard
Linköping, Sweden

Preface

Purpose

This is a hands-on guidebook on distributed simulation (DS), specifically on High Level Architecture (HLA), with a view toward software development. It contains a variety of examples to support learning by doing. It offers practical advice on real-world development issues for the novice engineers and programmers who are entering the field of distributed simulation.

Rationale

The book elaborates the implementation of an HLA federation covering all areas from the object model development to federate application development by exemplifying all the federate interface service areas. More importantly, the federate application development is based on the layered architectural style, which provides a clear separation of concerns. The implementation is based on the latest HLA standard, known as HLA Evolved, yet we point to the related topics with HLA 1.3 specification in the interest of backward compatibility. We also present a running example involving a great deal of variation to illustrate several practical issues of federate application development. The implementation is not based on a specific commercial runtime infrastructure (RTI) software, but rather based on generic and extendible tools that can be obtained freely. One is SimGe, a freeware, which is a fully dressed HLA object model development tool and a code generator. Second is RACoN, which is fundamentally an open-source .NET wrapper for HLA RTI application programming interface (API). The book serves as the only comprehensive reference for both tools by giving downloadable sample case studies with the source code. Moreover, this book contains a unique chapter that shows the practitioner how to employ HLA in (multi) agent-based simulation.

The prominent features are as follows:

– It comes with the tools RACoN and SimGe to enable the reader work at a conceptually high level.

- Numerous step-by-step examples and code snippets help the reader to understand the RTI concepts.
- It includes a running example involving a great deal of variation to illustrate several practical issues of federate application development.
- It includes a larger-scale case study involving multi-agents. The scenario is based on the maritime border surveillance with the help of a fleet of unmanned surface vehicles.
- It includes downloadable sample source code.
- It uses Microsoft .NET platform and the C# programming language for all implemented examples and case study.
- It provides a fast start-up on HLA federation development giving an implemented sample for each service area of the HLA federate interface specification.
- It includes a unique chapter to employ the HLA in multi-agent simulations.
- It covers not only federate application development, but also object model construction.
- Many chapters include questions for review and further study.

Book Overview

The book is structured as follows. Chapter 1 is a high-level introduction to the essential concepts of modeling and simulation, while highlighting DS as the focal area of interest. Chapter 2 introduces the fundamental concepts and the principles of HLA. Chapter 3 takes a look into federation development and presents development guidelines. These three chapters together lay the technical background for federation development.

The remaining chapters constitute three major parts of the book: object model development, federate implementation, and advanced topics. Chapters 4 and 5 present an introduction to object model development and then elaborate on the subject through a case study. Using the object model in the case study, Chaps. 6–8 elucidate the details of implementing a federate application. Last, Chap. 10 considers some advanced topics by discussing the connection between agent-based simulation and HLA. Then, Chap. 11 provides a complete case study to put it all together.

The chapters are structured in a layered way so that the initial chapters include more generalized topics such as M&S and HLA concepts. As we proceed, each chapter brings more specialized topics relying on the knowledge of the preceding ones.

Web Material

The software tools and examples used in the book are freely available on the web.

Audience

This book is intended for students of distributed simulation, based particularly on High Level Architecture. It can be used as a textbook or reference book for an upper undergraduate/lower graduate course, probably named as distributed simulation or distributed interactive simulation.

Final

We believe that the most prominent contribution of this book is to provide a starting point and all-in-one resource for HLA-based distributed simulation development without depending on any commercial tools. The book is complementary, in regard to its implementation slant, to model-based engineering approach for distributed simulations.

Edirne, Turkey Okan Topçu
Ankara, Turkey Halit Oğuztüzün

Acknowledgements

We would like to acknowledge our associate editor Simon Rees for smooth editorial assistance.

Okan would like to thank his parents Selime and Bekir for their boundless support.

Halit would like to thank all his graduate students, especially those who graduated from the MODSIM master of science program at METU.

For various chapters of this book, we have adapted parts of the following articles/chapters:

Topçu, O., Durak, U., Oğuztüzün, H. & Yılmaz, L., 2016. Distributed Simulation: A Model Driven Engineering Approach. 1st ed. Cham(Zug): Springer International Publishing. Parts adapted and reprinted with permission from Springer appear in Chaps. 1, 2 and 7.

Contents

Part II Object Model Development

Abbreviations

ACL	Agent Communication Language
AgentFdApp	Agent Manager Federate Application
ALSP	Aggregate Level Simulation Protocol
AMG	Architecture Management Group
AOR	Area of Responsibility
API	Application Programming Interface
ARPA	Advanced Research Projects Agency
C2	Command and Control
CGB	Coast Guard Boat
CGF	Computer Generated Force
CodeGen	Code Generator (of SimGe)
CogAgentLib	Coherence-based Agent Framework
COTS	Commercial off-the-shelf
CRUD	Create, read, update, and delete operations
D/A	Divest/Acquire
DDM	Data Distribution Management, Detailed Design Model
DeCoAgent	Deliberative Coherence Driven Agent
DIF	Data Interchange Format
DIS	Distributed Interactive Simulation
DLC	Dynamic Link Compatibility
DLL	Dynamic Link Library
DM	Declaration Management
DMAO	Distributed Simulation Engineering and Execution Process Multi-Architecture Overlay
DMSO	U.S. Defense Modeling and Simulation Office
DoD	U.S. Department of Defense
DOI	Digital Object Identifier
DSEEP	Distributed Simulation Engineering and Execution Process
DUB	Dimension Upper Bound
EnviFd	Environment Federate
EnviFdApp	Environment Controller Federate Application
FAM	Federation Architecture Model
FAME	Federation Architecture Modeling Environment

FAMM	Federation Architecture Metamodel
FAP	Federation Architecture Project
FDD	FOM Document Data
FED	Federation Execution Details
FEDEP	Federation Development and Execution Process
FedMonFd	Federation Monitor Federate
FedMonFdApp	Federation Monitor Federate Application
FFL	Federation Foundation Library
FIPA	Foundation for Intelligent Physical Agents
FM	Federation Management
FOM	Federation Object Model
fps	Frame per Second
GALT	Greatest Available Logical Time
GPS	Global Positioning System
GUI	Graphical User Interface
HLA	High Level Architecture
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IF	Interface Specification
IP	Internet Protocol
ISBN	International Standard Book Number
KQML	Knowledge Query and Manipulation Language
LITS	Least Incoming Timestamp
M&S	Modeling and Simulation
MariSim	Maritime Simulation
MCV	Model-View-Controller
MDE	Model-Driven Engineering
MIM	MOM and Initialization Module
MOM	Management Object Model
MRU	Most Recently Used
MS	Microsoft
MSF	Maritime Surveillance Federation
NavySim	Naval Simulation
NSTMSS	Naval Surface Tactical Maneuvering Simulation System (pronounced “Nistmiss”)
OM	Object Model, Object Management
OME	Object Model Editor (of SimGe)
OMG	Object Modeling Group
OMT	Object Model Template
OOP	Object Oriented Programming
OOW	Officer of the Watch
OwM	Ownership Management
P/S	Publish/Subscribe
PDU	Protocol Data Unit
POC	Point of Contact

RACoN	RTI Abstraction Component for .Net
RB	Refugee Boat
RID	RTI Initialization Data
RO	Receive Order
RPR-FOM	Real-time Platform Reference FOM
RTI	Runtime Infrastructure
ShipFd	Ship Federate
ShipFdApp	Ship Federate Application
SimGe	SIMulation Generator
SISO	Simulation Interoperability Standards Organization
SL	Semantics Language
SOM	Simulation Object Model
SS	Support Services
StationFd	Station Federate
StationFdApp	Station Federate Application
STMS	Strait Traffic Monitoring Simulation
T/A	Transferable/Acceptable
TC	Time-Constrained
TC&TR	Both TC and TR
TM	Time Management
TMS	Traffic Monitoring Station
TR	Time-Regulating
TSO	Time Stamp Order
TSS	Tracking Sub-System
U/R	Updateable/Reflectable
UI	User Interface
UML	Unified Modeling Language
URL	Uniform Resource Locator
US DoD	United States Department of Defense
USV	Unmanned Surface Vehicle
VS	Visual Studio
WiX	Windows Installer XML
WPF	Windows Presentation Foundation
XML	Extensible Markup Language