

IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Framework and Rules

Sponsor

Simulation Interoperability Standards Committee (SISC)
of the
IEEE Computer Society

Approved 21 September 2000

IEEE-SA Standards Board

Abstract: The High Level Architecture (HLA)—Framework and Rules is the capstone document for a family of related HLA standards. It defines the HLA, its components, and the rules that outline the responsibilities of HLA federates and federations to ensure a consistent implementation. Simulations are abstractions of the real world, and no one simulation can solve all of the functional needs for the modeling and simulation community. It is anticipated that technology advances will allow for new and different modeling and simulation (M&S) implementations within the framework of the HLA. The standards contained in this architecture are interrelated and need to be considered as a product set, as a change in one is likely to have an impact on the others. As such, the HLA is an integrated approach that has been developed to provide a common architecture for simulation.

Keywords: architecture, class attribute, federate, federation, federation execution, federation object model, framework, high level architecture, instance attribute, interaction class, joined federate, object class, object model template, rules, runtime infrastructure, simulation object model

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2000 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 11 December 2000. Printed in the United States of America.

Print: ISBN 0-7381-2619-5 SH94882
PDF: ISBN 0-7381-2620-9 SS94882

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

IEEE is the sole entity that may authorize the use of certification marks, trademarks, or other designations to indicate compliance with the materials set forth herein.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; (978) 750-8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

(This introduction is not part of IEEE Std 1516-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Framework and Rules.)

This standard is the capstone document for a family of related standards that together describe a unified approach to constructing interoperable simulation systems.

The HLA provides a general framework within which simulation developers can structure and describe their simulation applications. Flexibility is the aim of the HLA. In particular, the HLA addresses two key issues: promoting interoperability between simulations and aiding the reuse of models in different contexts. Two main components are described within the set of products forming the HLA. The first is the Object Model Template (OMT), which forms a documentation standard describing the data used by a particular model, a necessary basis for reuse. The second component, the Federate Interface Specification, describes a generic communications interface that allows simulation models to be connected and coordinated, thus, addressing interoperability. Although the HLA is an architecture, not software, use of runtime infrastructure (RTI) software is required to support operations of a federation execution. The RTI software provides a set of services, as defined by the Federate Interface Specification, used by federates to coordinate operations and data exchange during a runtime execution.

Simulations are necessarily abstractions of the real world, and no one simulation design can meet the functional needs of the entire modeling and simulation community. However, in defining an overriding architecture, generic issues can be addressed. When doing so, it is essential that such an architecture encompass both differing computing environments and differing classes of simulations.

This standard document of the Framework and Rules is intended to provide some of the general philosophy behind the HLA, including guidance about how to design, use, and adhere to the HLA vision.

Participants

This standard was prepared by the HLA Working Group, sponsored by the Simulation Interoperability Standards Committee (SISC) of the IEEE Computer Society. When the 1516 working group approved this standard, it had the following membership:

Susan Symington, *Chair*
Katherine L. Morse, *Vice Chair*
Mikel Petty, *Secretary*

Stephen Bachinsky
Joe Batman
Brian Beebe
Keith Briggs
Jim Calvin
Richard M. Fujimoto
Deborah Fullford
Allison Griffin

James H. Hammond
James Ivers
Andreas Kemkes
John F. Kramer
Frederick S. Kuhl
Gary M. Lightner
Reed Little
Rodney Long
Margaret Loper
Robert R. Lutz

Jeffrey M. Nielsen
Marnie R. Salisbury
Randy Saunders
Roy Scrudder
David W. Seidel
Robert C. Turrell
Richard Weatherly
Annette Wilson

The working group acknowledges the following individuals who also contributed to the preparation of this standard:

Phillippe Annic
Tobin Bergen-Hill
Dominique Canazzi

Judith S. Dahmann
Peter Hoare
Mikael Karlsson

Michael Mazurek
J. Russell Noseworthy
Graham Shanks

The following members of the balloting committee voted on this standard:

Dale E. Anglin
Stephen Bachinsky
Michael R. Bachmann
David L. Barton
Brian Beebe
Robert P. Bennett
Christina L. Bouwens
Keith Briggs
Richard Briggs
Danny Cohen
Glenn Conrad
Mark Crooks
Dannie Cutts
Judith S. Dahmann
Steven Drake
Laura E. Feinerman
Jeff E. Fischer
Deborah Fullford
Richard M. Fujimoto
Brian F. Goldiez
Victor M. Gonzalez
James H. Hammond
Jack G. Harrington
Robert V. Head, Jr
Callie M. Hill
Ronald C. Hofer
James W. Hollenbach
Margaret M. Horst

Robert J. Howard
Ken Hunt
Kyle Isakson
James Ivers
Andreas Kemkes
John F. Kramer
Frederick S. Kuhl
Tom Lake
Gary M. Lightner
Paul R. Little
Reed Little
Margaret Loper
Robert R. Lutz
Jayne Lyons
Theodore F. Marz
Mark McAuliffe
David R. McKeeby
Duncan C. Miller
Katherine L. Morse
Thomas H. Mullins
Jeffrey M. Nielsen
Michael J. O'Connor
John Peng
Paul Perkinson
Hung Q. Phan
David R. Pratt
Paul M. Reeves
David Roberts
Marnie R. Salisbury
Randy Saunders

Roy Scrudder
David W. Seidel
Richard J. Severinghaus
Joseph F. Seward
Sean Sharp
Steven M. Sheasby
John W. Sheppard
Allen H. Skees
Col. Mark Smith
Susan D. Solick
Sandra Swearingen
Stephen J. Swenson
Susan F. Symington
Donald Theune
William V. Tucker
Grant R. Tudor
John A. Tufarolo
Robert C. Turrell
Andrew J. Ventre
William F. Waite
Charles E. Walters
Richard Weatherly
Chris C. Wertman
Annette Wilson
Robert Whittman
Doug Wood
Philomena M. Zimmerman
William H. Zimmerman

When the IEEE-SA Standards Board approved this standard on 21 September 2000, it had the following membership:

Donald N. Heirman, *Chair*
James T. Carlo, *Vice Chair*
Judith Gorman, *Secretary*

Satish K. Aggarwal
Mark D. Bowman
Gary R. Engmann
Harold E. Epstein
H. Landis Floyd
Jay Forster*
Howard M. Frazier
Ruben D. Garzon

James H. Gurney
Richard J. Holleman
Lowell G. Johnson
Robert J. Kennelly
Joseph L. Koepfinger*
Peter H. Lips
L. Bruce McClung
Daleep C. Mohla

James W. Moore
Robert F. Munzner
Ronald C. Petersen
Gerald H. Peterson
John B. Posey
Gary S. Robinson
Akio Tojo
Donald W. Zipse

*Member Emeritus

Also included is the following nonvoting IEEE-SA Standards Board liaison:

Alan Cookson, *NIST Representative*
Donald R. Volzka, *TAB Representative*

Jennifer McClain Longman
IEEE Standards Project Editor

Contents

1. Overview.....	1
1.1 Scope.....	1
1.2 Purpose.....	1
1.3 Relationship of HLA and object-oriented concepts.....	1
2. References.....	2
3. Definitions, abbreviations and acronyms.....	2
3.1 Definitions	2
3.2 Abbreviations and acronyms	14
4. Summary of the HLA rules.....	15
5. Federation Rules	15
5.1 Rule 1	15
5.2 Rule 2.....	15
5.3 Rule 3.....	16
5.4 Rule 4.....	16
5.5 Rule 5.....	16
6. Federate rules.....	17
6.1 Rule 6.....	17
6.2 Rule 7.....	17
6.3 Rule 8.....	17
6.4 Rule 9.....	17
6.5 Rule 10.....	17
Annex A (informative) Rationale.....	18
Annex B (informative) Bibliography	21

IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Framework and Rules

1. Overview

1.1 Scope

This document provides an overview of the High Level Architecture (HLA), defines a family of related HLA documents, and defines the principles of HLA in terms of responsibilities that federates (simulations, supporting utilities, or interfaces to live systems) and federations (sets of federates working together) must uphold.

1.2 Purpose

This document describes the general principles defining the HLA and delineates the set of rules that apply to HLA federations and federates. Each rule is then described, and the rationale for its inclusion is provided.

Many different classes of simulations exist. Each class has changing application characteristics and needs to be flexibly supported to allow for interoperability and reuse across classes and to limit the need to maintain multiple interoperability approaches. The HLA is an integrated architecture, which has been developed to provide that common architecture. The related standards need to be considered as a set of products because changes in one are likely to have an impact on the others.

1.3 Relationship of HLA and object-oriented concepts

Although the HLA Object Model Template (OMT) is the standardized documentation structure for HLA object models, Federation Object Models (FOMs) and Simulation Object Models (SOMs) do not completely correspond to common definitions of object models in object-oriented (OO) analysis and design (OOAD) techniques. In the OOAD literature, an object model is described as an abstraction of a system developed for the purpose of fully understanding the system. To achieve this understanding, most OO techniques recommend defining several views of the system. For HLA object models, the intended scope of the system description is much narrower, focusing specifically on requirements and capabilities for federate information exchange. For SOMs, the intent is to describe the public interface of the federate in terms of an

identified set of supported HLA object classes and interaction classes. A more complete description of how a federate is designed and functions internally (for example, a traditional OO object model) should be provided via documentation resources other than the SOM. For FOMs, the intent is to describe information exchange that happens during a federation execution.

Differences between HLA and OOAD principles and concepts also appear at the individual object level. In the OOAD literature, objects are defined as software encapsulations of data and operations (methods). In the HLA, objects are defined entirely by the identifying characteristics (attributes), values of which are exchanged between federates during a federation execution. Any OO-related behaviors and operations that affect the values of HLA object attributes are kept resident in the federates. Although HLA class structures are driven by subscription requirements and FOM growth concerns, class structures in OO systems are typically driven by efficiency and maintainability concerns.

As discussed above, HLA object classes are described by the attributes that are defined for them. These are, in OO parlance, data members of the class. These attributes, which are abstract properties of HLA object classes, are referred to as class attributes. HLA object instances are spawned via an HLA service using an HLA object class as a template. Each attribute contained by an HLA object instance is called an instance attribute.

OO objects interact via message passing, in which one OO object invokes an operation provided by another OO object. HLA objects do not directly interact. It is the federates that interact, via HLA services, by updating instance attribute values or sending interactions. Also, responsibility for updating the instance attributes associated with an HLA object instance can be distributed among different federates in a federation (effectively distributing responsibility for maintaining the HLA object instance's state across the federation), whereas OO objects encapsulate state locally and associate update responsibilities with operations that are closely tied to the object's implementation in an OO programming language.

In addition to the stated semantic variations in shared terminology, other differences may also exist. Precise definitions of all HLA terms can be found in Clause 3.

2. References

This standard shall be used in conjunction with the following publications. When the following specifications are superseded by an approved revision, the revision shall apply:

IEEE 1516.1-2000, IEEE Standard for Modeling and Simulation (M&S) High-Level Architecture (HLA)—Federate Interface Specification.¹

IEEE 1516.2-2000, IEEE Standard for Modeling and Simulation (M&S) High-Level Architecture (HLA)—Object Model Template (OMT) Specification.

3. Definitions, abbreviations and acronyms

3.1 Definitions

For the purposes of this standard, the following terms and definitions apply. *The Authoritative Dictionary of IEEE Standards Terms* [B1]² should be referenced for terms not defined in this clause.

¹IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA (<http://standards.ieee.org/>).

²The numbers in brackets correspond to those of the bibliography in Annex B.

3.1.1 accuracy: The measure of the maximum deviation of an attribute or a parameter value in the simulation or federation from reality or some other chosen standard or referent.

3.1.2 active subscription: A request to the runtime infrastructure (RTI) for the kinds of data (class attributes as well as interactions) that the joined federate is currently interested in receiving. The RTI uses this data along with publication data received from other joined federates to support services, such as

- a) *Start/Stop Registration*
- b) *Turn On/Off Updates*
- c) *Turn On/Off Interactions*

The RTI also uses the subscription (and publication) data to determine how to route data to the joined federates that require that data.

3.1.3 attribute: A named characteristic of an object class or object instance. *See also:* **class attribute** and **instance attribute**.

3.1.4 attribute ownership: The property of an instance attribute that gives a joined federate the capability to supply values for that instance attribute to its federation execution. *See also:* **instance attribute**.

3.1.5 available attributes: The set of declared attributes of an object class in union with the set of inherited attributes of that object class.

3.1.6 available dimensions:

- a) Pertaining to an attribute: the dimensions associated with the class attribute in the Federation Object Model (FOM). The available dimensions of an instance attribute are the available dimensions of the corresponding class attribute.
- b) Pertaining to an interaction class: the dimensions associated with that interaction class in the FOM. The available dimensions of a sent interaction are the available dimensions of the interaction class specified in the *Send Interaction With Regions* service invocation.

NOTE—See 9.1 in IEEE Std 1516.1-2000.

3.1.7 available parameters: The set of declared parameters of an interaction class in union with the set of inherited parameters of that interaction class.

3.1.8 candidate discovery class: The registered class of an object instance, if subscribed. If the registered class of an object instance is not subscribed, the closest superclass of the registered class of the object instance to which the joined federate is subscribed. The term candidate discovery class pertains to object instances only.

3.1.9 candidate received class: The sent class of an interaction, if subscribed. If the sent class of an interaction is not subscribed, the closest superclass of the sent class of the interaction to which the joined federate is subscribed. The term candidate received class pertains to interactions only.

3.1.10 class: A description of a group of items with similar properties, common behavior, common relationships, and common semantics.

3.1.11 class attribute: A named characteristic of an object class denoted by a pair composed of an object class designator and an attribute designator.