

IEEE Guide to Software Design Descriptions

IEEE Standards Board

Approved March 18, 1993

American National Standards Institute

Approved August 30, 1993

Sponsor

Software Engineering Standards Subcommittee of the Technical Committee
on Software Engineering
of the
IEEE Computer Society

Abstract: The application of design methods and design documentation recommended in IEEE Std 1016-1987 is described. Several common design methods are used to illustrate the application of IEEE Std 1016-1987, thus making the concepts of that standard more concrete. The information in this guide may be applied to commercial, scientific, or military software that runs on any computer. Applicability is not restricted by the size, complexity, or criticality of the software.

Keywords: design entity, design method, design view, software design process.

The Institute of Electrical and Electronics Engineers, Inc. 345 East 47th Street, New York, NY 10017-2394, USA

Copyright © 1993 by the Institute of Electrical and Electronics Engineers, Inc. All rights reserved. Published 1993. Printed in the United States of America

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 Technical Committees of the IEEE Societies and the Standards Coordinating Committees of the IEEE 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 technical 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 Standards Board 445 Hoes Lane P.O. Box 1331 Piscataway, NJ 08855-1331 USA

IEEE Standards documents are adopted by the Institute of Electrical and Electronics Engineers without regard to whether their adoption may involve patents on articles, materials, or processes. Such adoption does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the standards documents.

Introduction

(This introduction is not a part of IEEE Std 1016.1-1993, IEEE Guide to Software Design Descriptions.)

Audience

This guide is intended for those in technical and managerial positions who prepare and use software design descriptions. It will guide in understanding IEEE Std 1016-1987 and in the selection, organization, and presentation of design information. This guide will also help standards developers to apply IEEE Std 1016-1987 in preparing design documentation standards.

At the time this standard was completed, the Software Design Description Working Group had the following membership:

	Basil Sherlund, <i>Chair</i>	
Don Chandler	John McArdle	Roger Van Scoy
Francois Coallier	Diane Mularz	John Wilson
Pat Finnigan	Ilona Rossman	Rodger Zambis
	David Spence	

In the final stage of revision, responses to ballot comments was provided by:

	Basil Sherlund, <i>Chair</i>	
William Hefley		Vladan Jovanovic

The following persons were on the balloting committee that approved this document for submission to the IEEE Standards Board:

G. Arato	H. Harauz	S. R. Schach
D. Avery	W. Hefley	G. D. Schumacher
J. H. Barnard	J. W. Horch	R. W. Shillato
R. Berlack	M. S. Karasik	D. M. Siefert
F. Buckley	R. Kosinski	H. M. Sneed
G. Cozens	T. Kurihara	A. Sorkowitz
B. K. Derganc	R. Lane	V. Srivastava
E. Dragstedt	B. Lau	R. H. Thayer
J. W. Fendrich	B. Livson	G. D. Tice
K. Fortenberry	J. Maayan	L. L. Tripp
J. Garbajosa	J. Marijarvi	R. Van Scoy
D. Gelperin	R. Martin	D. Wallace
Y. Gershkovitch	S. C. Mathews	W. M. Walsh
J. Gonzalez	I. Mazzo	P. R. Work
D. A. Gustafson	S. Mcgrath	J. Zalweski
	D. E. Nickle	

When the IEEE Standards Board approved this standard on March 18, 1993, it had the following membership:

	Wallace S. Read, <i>Chair</i>	
	Donald C. Loughry, <i>Vice Chair</i>	
	Andrew G. Salem, <i>Secretary</i>	
Gilles A. Baril	Ben C. Johnson	Marco W. Migliaro
Clyde R. Camp	Walter J. Karplus	L. John Rankine
Donald C. Fleckenstein	Lorraine C. Kevra	Arthur K. Reilly
Jay Forster*	E. G."Al" Kiener	Ronald H. Reimer
David F. Franklin	Ivor N. Knight	Gary S. Robinson
Ramiro Garcia	Joseph L. Koepfinger*	Leonard L. Tripp
Donald N. Heirman	D. N."Jim" Logothetis	Donald W. Zipse
Jim Isaak	Don T. Michael*	

*Member emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal	Richard B. Engelman	Stanley Warshaw
James Beall	David E. Soffrin	
	Rachel Auslander, <i>Editor</i>	

Contents

CLAUSE	PAGE
1. Overview	1
1.1 Purpose.....	1
1.2 Scope.....	1
2. References.....	1
3. Definitions.....	2
4. Description of IEEE Std 1016-1987	3
5. Design description organization	3
5.1 Design views	3
5.2 Recommended design views.....	4
5.3 Design description media.....	4
6. Considerations.....	5
6.1 Selecting representative design methods	5
6.2 Representative design method descriptions	5
6.3 Design document sections.....	6
6.4 Method-oriented design documents	6
7. Design methods.....	7
7.1 Function-oriented design methods	7
7.2 Data-oriented design methods.....	8
7.3 Real-time control-oriented design methods	9
7.4 Object-oriented design methods	10
7.4 Object-oriented design methods	10
7.5 Formal language-oriented design methods	12
8 Bibliography	13
ANNEX	
Annex (informative) Selecting a software design notion	15
A.1 Data flow diagrams (DFDs).....	15
A.2 Transformation graphs (TRGs).....	15
A.3 State transition diagrams (STDs)	16
A.4 Structure charts (STCs).....	16
A.5 Entity relationship diagrams (ERDs)	17
A.6 Objects	18

IEEE Guide to Software Design Descriptions

1. Overview

This guide is divided into eight clauses and has one annex. Clause 1 provides the purpose and scope of this guide. Clause 2 lists references to other standards that are useful to applying this guide. Clause 3 provides definitions of terms used in this guide. Clause 4 briefly describes IEEE Std 1016-1987. Clause 5 provides the context for this guide. Clause 6 provides the considerations regarding this guide and introduces the concept of a design description. Clause 7 provides a description of common design methods and their relationships to IEEE Std 1016-1987. Clause 8 contains a bibliography. The annex, although not part of this guide, provides a basis for selecting a design method.

1.1 Purpose

The purpose of this guide is to show how to apply IEEE Std 1016-1987, IEEE Recommended Practice for Software Design Descriptions. The approach used to do this is as follows: The relationship between IEEE Std 1016-1987 and design methods that are familiar and widely used is shown. This includes identifying what design entities each method recognizes and what entity attributes it emphasizes. The set of models (or views) that each design method requires are also identified.

1.2 Scope

This guide describes the application of design methods and design documentation recommended in IEEE Std 1016-1987. Since most design methods specify media and information, organization of the material is left up to the user of the design method. This guide shows how the information requirements of IEEE Std 1016-1987 are met by several common design methods. IEEE Std 1016-1987 is not limited to the specific methods described in this guide. Rather, these examples were selected to illustrate the application of IEEE Std 1016-1987. The methods selected as examples are widely used and are familiar, thus making the concepts of IEEE Std 1016-1987 more concrete.

This information in this guide may be applied to commercial, scientific, or military software that runs on any computer. Applicability is not restricted by the size, complexity, or criticality of the software.

2. References

This guide shall be used in conjunction with the following publications:

IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology (ANSI).¹

IEEE Std 1016-1987, IEEE Recommended Practice for Software Design Descriptions (ANSI).

¹ IEEE publications are available from the Institute of Electrical and Electronics Engineers, Service Center, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA.

3. Definitions

This clause contains key terms as they are used in this guide. Definitions of other terms used in this guide can be found in IEEE Std 610.12-1990.²

3.1 component: A distinct part of a subsystem. A component may be decomposed into other components and computer software units.

3.2 design description: A document that describes the design of a system or component. Typical contents include system or component architecture, control logic, data structures, input/output formats, interface descriptions, and algorithms.

3.3 design entity: An element (component) of a design that is structurally and functionally distinct from other elements and that is separately named and referenced.

3.4 design method: A definition of a set of essential entities. It includes a systematic procedure for the set of models with rules or heuristics used to determine the entities and entity attributes of the models, and a notation to represent the entities and the attributes expressed in each model.

3.5 design methodology: A guideline identifying how to design software. As a process, a methodology is a practical set of procedures that facilitate the design of software.

3.6 design view: A subset of design entity attribute information that is specifically suited to the needs of a software project activity.

3.7 entity attribute: A named characteristic or property of a design entity that provides a systematic procedure for the statement of fact about the entity.

3.8 model: A representation of one or more aspects of a system.

3.9 notation: A set of symbols used to represent design entities and entity attributes.

3.10 software configuration item, or subsystem: A collection of software elements treated as a unit for the purpose of configuration management.

3.11 software design description (SDD): A representation of a software system created to facilitate analysis, planning, implementation, and decision making. It is a blueprint or model of the software system. The SDD is used as the primary medium for communicating software design information.

3.12 software design document: The output of design process in a presentable format, traditionally, a paper-based document.

3.13 software design process: Organized tasks and activities of design, having appropriate specification.

3.14 software design process specification: Know-how, technology of design, that specify operationally how to use methodology of design (standardized itself) together with standards for evaluating design, tools to support design automation, and documentation required to represent design information.

² Information on references can be found in clause 2 .

4. Description of IEEE Std 1016-1987

IEEE Std 1016-1987 is an information standard (that is, it specifies the information that a design method should contain), rather than a documentation standard (that is, a standard that specifies format). It prescribes the minimum information requirements that must be met by a design description, but it does not prescribe an organization for that information in a document.

Section 5 of IEEE Std 1016-1987 introduces the notion of a design entity and indicates the minimum set of design entity attributes that shall be specified in a design description. These attributes include the following:

- a) Identification
- b) Type
- c) Purpose
- d) Function
- e) Subordinates
- f) Dependencies
- g) Interface
- h) Resources
- i) Processing
- j) Data

Section 6 of IEEE Std 1016-1987 introduces the notion of a design view. A design view is a subset of design entity attribute information that is specifically suited to the needs of a software project activity. Design views may provide subsets of design entity attribute information such as hierarchy, control flow, data flow, data structure, time dependent behavior, or mode/state transition control. Partial design descriptions, such as those presented in sections of a design document, are design views. They describe something about the system but not the entire design. IEEE Std 1016-1987 provides a set of four common views. They are the following:

- a) Decomposition
- b) Dependency
- c) Interface
- d) Detailed design

These views are not required to meet the recommended practices in IEEE Std 1016-1987. Rather, they are introduced to illustrate the concept of views. Different design methods will usually have a different set of design views that are considered important.

5. Design description organization

This clause discusses the organization of information into design views, recommended design views, and the various media that can be used to illustrate a design description.

5.1 Design views

A design view is a means of organizing design information. It is a subset of entity attribute information. A complete design description provides all attribute information about all entities. However, a design view looks at only the following:

- a) *A subset of the design entities.* In this situation the design view gives all design information about part of the system.