

IEEE Guide for Developing System Requirements Specifications

Sponsor

**Software Engineering Committee
of the
IEEE Computer Society**

Approved 17 April 1996

IEEE Standards Board

Approved 3 March 1997

American National Standards Institute

Abstract: Guidance for the development of the set of requirements, System Requirements Specification (SyRS), that will satisfy an expressed need is provided. Developing a SyRS includes the identification, organization, presentation, and modification of the requirements. Also addressed are the conditions for incorporating operational concepts, design constraints, and design configuration requirements into the specification. This guide also covers the necessary characteristics and qualities of individual requirements and the set of all requirements.

Keywords: requirement, system, system requirements specifications

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

Copyright © 1996 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1996. Printed in the United States of America.

ISBN 1-55937-716-X

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

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 all 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.

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; (508) 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 a part of IEEE Std 1233-1996, IEEE Guide for Developing System Requirements Specifications.)

The purpose of this guide is to provide guidance for capturing system requirements. This guide serves the analyst by providing a clear definition for identifying well-formed requirements and ways of organizing them.

This guide should be used to help the analyst capture requirements at the beginning of the system requirements phase. It should be used to clarify what constitutes a good requirement and provide an understanding of where to look to identify different requirement sources.

This guide was prepared by a working group chartered by the Software Engineering Committee of the Computer Society of IEEE. This guide represents a consensus of individual working-group participants with broad expertise in systems engineering and requirements analysis.

The following individuals contributed to the writing of this guide by attending two or more working sessions, by substantial written commentary, or both.

At the time this guide was completed, the Working Group had the following membership:

Louis E. Miller, *Chair*
William N. Sabor, *Secretary*

Bakul Banerjee
David Byrch
Kim A. Cady
Larry Diehr
Charles A. Droz
Larry C. Forrest

P. Michael Guba
James R. Hughes
Joe Iaquinto
Marybeth A. Jupina
Thomas M. Kurihara
Richard C. Lee

Jim Longbucco
Donald F. Parsons
Eric Peterson
John Sheckler
Jess Thompson
Eva D. Williams

Other contributors include:

Geoff Cozens
Paul Davis

Kristin Dittmann
Christof Ebert
Don McCash

Virginia Nuckolls
Anne O'Neill

The following persons were on the balloting committee:

H. Ronald Berlack
Mark Bilger
William J. Boll
Fletcher Buckley
Edward R. Byrne
Francois Coallier
Christopher Cooke
Geoff Cozens
Alan M. Davis
Robert E. Dwyer
Sherman Eagles
Leo G. Egan
Caroline L. Evans
Richard L. Evans
John W. Fendrich
Peter Fillery
Larry Forrest
Eugene P. Friedman
Eitan Froumine
Yair Gershkovich
Adel N. Ghannam
Julio Gonzalez Sanz

Patrick J. Griffin
David A. Gustavson
John Harauz
Derek J. Hatley
William Hefley
Umesh P. Hiriyannaiah
Jody Howard
Eiichi Kaneko
Jerry Kickenson
Janet Kintner
Thomas M. Kurihara
Renee Lamb
John B. Lane
Boniface Lau
J. Dennis Lawrence
Ben Livson
Harold Mains
Roger Martin
James W. Mc Clean
Sue McGrath
Louis E. Miller
Dennis E. Nickle

Indradeb P. Pal
Joseph A. Palermo
Stephen R. Schach
Norman Schneidewind
Wolf A. Schnoege
Gregory D. Schumacher
Carl S. Seddio
David M. Seifert
Richard S. Sky
Alfred R. Sorkowitz
Robert N. Sulgrove
Tanehiro Tatsuta
Richard H. Thayer
George D. Tice
Leonard L. Tripp
Tom Vaiskunas
Thomas E. Vollman
Ronald L. Wade
Dolores Wallace
William M. Walsh
Paul R. Work
Janusz Zalewski

When the IEEE Standards Board approved this standard on 17 April 1996, it had the following membership:

Donald C. Loughry, *Chair*

Richard J. Holleman, *Vice Chair*

Andrew G. Salem, *Secretary*

Gilles A. Baril
Clyde R. Camp
Joseph A. Cannatelli
Stephen L. Diamond
Harold E. Epstein
Donald C. Fleckenstein
Jay Forster*
Donald N. Heirman

Ben C. Johnson
E. G. "Al" Kiener
Joseph L. Koepfinger*
Lawrence V. McCall
L. Bruce McClung
Marco W. Migliaro
Mary Lou Padgett
John W. Pope
Jose R. Ramos

Arthur K. Reilly
Ronald H. Reimer
Gary S. Robinson
Ingo Rüsçh
John S. Ryan
Chee Kiow Tan
Leonard L. Tripp
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal
Alan H. Cookson
Chester C. Taylor

Angela M. Girardi
IEEE Standards Project Editor

Contents

CLAUSE	PAGE
1. Overview.....	1
1.1 Scope.....	1
2. References.....	1
3. Definitions	2
4. System requirements specification	4
4.1 Definition	4
4.2 Properties	4
4.3 Purpose.....	5
4.4 Intended use	6
4.5 Benefits	6
4.6 Dynamics of system requirements	7
5. SyRS development process overview	7
5.1 Customer	7
5.2 Environment.....	8
5.3 Technical community	10
6. Well-formed requirements	11
6.1 Definition of well-formed requirement.....	11
6.2 Properties of a requirement.....	13
6.3 Categorization	13
6.4 Pitfalls	14
7. SyRS development.....	15
7.1 Identify requirements	15
7.2 Build a well-formed requirement.....	17
7.3 Organize requirements	18
7.4 Present requirements.....	19
Annex A (informative) An SyRS outline	20
Annex B (informative) Bibliography.....	24

IEEE Guide for Developing System Requirements Specifications

1. Overview

1.1 Scope

This standard provides guidance for the development of a set of requirements that, when realized, will satisfy an expressed need. In this standard the set of requirements will be called the System Requirements Specification (SyRS). Developing an SyRS includes the identification, organization, presentation, and modification of the requirements. This guide addresses conditions for incorporating operational concepts, design constraints, and design configuration requirements into the specification. This guide also addresses the necessary characteristics and qualities of individual requirements and the set of all requirements.

This guide does not specify industry wide system specification standards nor state mandatory system requirements specification. This guide is written under the premise that the current state of the art of system development does not warrant or support a formal standards document.

2. References

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

ASTM E 623-89, Standard Guide for Developing Functional Requirements for Computerized Systems.¹

IEEE Std 100-1992, The New IEEE Standard Dictionary of Electrical and Electronics Terms (ANSI).²

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

IEEE Std 730-1989, IEEE Standard for Software Quality Assurance Plans (ANSI).

IEEE Std 828-1990, IEEE Standard for Software Configuration Management Plans (ANSI).

¹ASTM publications are available from the American Society for Testing and Materials, 100 Barr Harbor Drive, West Conshohocken, PA 19428-2959, USA.

²IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA.

IEEE Std 830-1993, IEEE Guide for Software Requirements Specifications (ANSI).

IEEE Std 1074-1995, IEEE Standard for Developing Software Life Cycle Processes (ANSI).

IEEE Std 1220-1994, IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process.

ISO 9000: 1994, Quality management and quality assurance standards: Guidelines for selection and uses.³

ISO 9126: 1991, Information tech. software system evaluation, Quality characteristics and guidelines for their use.

MIL-STD-490/DI-CMAN-80008A/DI-CMAN-80008, Military Standard Specification Practices.⁴

MIL-STD 498, Military Standard Defense System Software Development.

3. Definitions

The definitions listed below establish meaning in the context of this guide. Terms not defined in this standard are included in IEEE Std 610.12-1990.⁵

3.1 analyst: A member of the technical community (such as a systems engineer or business analyst, developing the system requirements) who is skilled and trained to define problems and to analyze, develop, and express algorithms.

3.2 annotation: Further documentation accompanying a requirement such as background information and/or descriptive material.

3.3 baseline: A specification or system that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development and can be changed only through formal change control procedures (IEEE 610.12-1990).

3.4 constraint: A statement that expresses measurable bounds for element or function of the system. That is, a constraint is a factor that is imposed on the solution by force or compulsion and may limit or modify the design changes.

3.5 customer(s): The entity or entities for whom the requirements are to be satisfied in the system being defined and developed. This can be an end-user of the completed system, an organization within the same company as the developing organization (e.g., System Management), a company or entity external to the developing company, or some combination of all of these. This is the entity to whom the system developer must provide proof that the system developed satisfies the system requirements specified.

3.6 derived requirement: A requirement deduced or inferred from the collection and organization of requirements into a particular system configuration and solution.

3.7 element: A component of a system; may include equipment, a computer program, or a human.

³ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembe, CH-1211, Genève 20, Switzerland/Suisse. ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

⁴MIL publications are available from Customer Service, Defense Printing Service, 700 Robbins Ave., Bldg. 4D, Philadelphia, PA 19111-5094.

⁵Information about references can be found in clause 2.

3.8 end user: The person or persons who will ultimately be using the system for its intended purpose.

3.9 environment: The circumstances, objects, and conditions that will influence the completed system; they include political, market, cultural, organizational, and physical influences as well as standards and policies that govern what the system must do or how it must do it.

3.10 function: A task, action, or activity that must be accomplished to achieve a desired outcome.

3.11 model: A representation of a real world process, device, or concept.

3.12 prototype: An experimental model, either functional or non functional, of the system or part of the system. A prototype is used to get feedback from users for improving and specifying a complex human interface, for feasibility studies, or identifying requirements.

3.13 raw requirement: An environmental or customer requirement that has not been analyzed and formulated as a well-formed requirement.

3.14 representation: A likeness, picture, drawing, block diagram, description, or symbol that logically portrays a physical, operational, or conceptual image or situation.

3.15 requirement: (a) A condition or capability needed by a user to solve a problem or achieve an objective. (b) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, a standard, specification, or other formally imposed documents. (c) A documented representation of a condition or capability as in (a) or (b). (IEEE Std 610.12-1990)

3.16 system: An interdependent group of people, objects, and procedures constituted to achieve defined objectives or some operational role by performing specified functions. A complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services, and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment.

3.17 System Requirements Specification (SyRS): The System Requirements Specification consists of a structured collection of information that embodies the requirements of the system.

3.18 testability: The degree to which a requirement is stated in terms that permit establishment of test criteria and performance of tests to determine whether those criteria have been met. (IEEE Std 610.12-1990)

3.19 traceability: The degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another; for example, the degree to which the requirements and design of a given system element match. (IEEE 610.12-1990)

3.20 validation: The process of evaluating a system or component during or at the end of the development process to determine whether a system or component satisfies specified requirements. (IEEE Std 610.12-1990)

3.21 verification: The process of evaluating a system or component to determine whether the system of a given development phase satisfy the conditions imposed at the start of that phase. (IEEE Std 610.12-1990)

3.22 well-formed requirement: A statement of system functionality (a capability) that can be validated, must be met or possessed by a system to solve a customer problem or to achieve a customer objective, and is qualified by measurable conditions and bounded by constraints.

4. System requirements specification

A System Requirements Specification (SyRS) has traditionally been viewed as a document that communicates the requirements of the customer to the technical community who will specify and build the system. The collection of requirements that constitutes the specification and its representation acts as the bridge between the two groups and must be understandable by both the customer and the technical community. One of the most difficult tasks in the creation of a system is that of communicating to all of the subgroups within both groups, especially in one document. This type of communication generally requires different formalisms and languages.

4.1 Definition

The SyRS presents the results of the definition of need, the operational concept, and the system analysis tasks. As such, it is a description of what the system's customers expect it to do for them, the system's expected environment, the system's usage profile, its performance parameters, and its expected quality and effectiveness. Thus it presents the conclusions of the SyRS Development process described in clause 5.

This guide suggests a distinction between this structured collection of information and the way in which it is presented to its various audiences. The presentation of the SyRS should take a form that is appropriate for its intended use. This can be a paper document, models, prototypes, other non-paper document representations, or any combination. All of these representations can be derived from this one SyRS to meet the needs of a specific audience. However, care should be taken to ensure that each of these presentations be traceable to a common source of system requirements information. The audience should be made aware that this structured collection of information remains the one definitive source for resolving ambiguities in the particular presentation chosen.

This guide makes a clear distinction between the system requirements (what the system must do) contained in the SyRS and process requirements (how to construct the system) that should be contained in contract documents like a Statement of Work.

4.2 Properties

The collection of requirements should have the following properties:

- a) Unique Set: Each requirement should be stated only once.
- b) Normalized: Requirements should not overlap, that is, they shall not refer to other requirements or the capabilities of other requirements.
- c) Linked Set: Explicit relationships should be defined among individual requirements to show how the requirements are related to form a complete system.
- d) Complete: An SyRS should include all the requirements identified by the customer, as well as those needed for the definition of the system.
- e) Consistent: SyRS content should be consistent and non contradictory in the level of detail, style of requirement statements, and in the presentation of material.
- f) Bounded: The boundaries, scope, and context for the set of requirements should be identified.
- g) Modifiable: The SyRS should be modifiable. Clarity and non overlapping requirements contribute to this.
- h) Configurable: Versions should be maintained versions time and across instances of the SyRS.
- i) Granular: This should be the level of abstraction for the system being defined.