

IEEE/EIA 12207.0-1996

(A Joint Standard Developed by IEEE and EIA)

IEEE/EIA Standard

Industry Implementation of International Standard ISO/IEC 12207 : 1995

(ISO/IEC 12207) Standard for Information Technology—

Software life cycle processes

March 1998



IEEE/EIA 12207.0-1996

(A Joint Standard Developed by IEEE and EIA)

IEEE/EIA Standard

Industry Implementation of International Standard ISO/IEC 12207: 1995

(ISO/IEC 12207) Standard for Information Technology—

Software life cycle processes

March 1998

Abstract: ISO/IEC 12207 provides a common framework for developing and managing software. IEEE/EIA 12207.0 consists of the clarifications, additions, and changes accepted by the Institute of Electrical and Electronics Engineers (IEEE) and the Electronic Industries Alliance (EIA) as formulated by a joint project of the two organizations. IEEE/EIA 12207.0 contains concepts and guidelines to foster better understanding and application of the standard. Thus this standard provides industry a basis for software practices that would be usable for both national and international business.

Keywords: acquisition process, audit, configuration management, development process, maintenance process, operation process, quality assurance, supply process, tailoring process, validation, verification

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

Print: ISBN 1-55937-977-4, SH94581
PDF: ISBN 0-7381-0428-0, SS94581

Copyright © 1998 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1998. 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.

March 1998

NOTICE

IEEE and EIA Standards and Publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Use of such Standards and Publications is wholly voluntary. Existence of such standards and Publications shall not in any respect preclude any member or nonmember of IEEE and EIA from manufacturing or selling products not conforming to such Standards and Publications, nor shall the existence of such Standards and Publications preclude their voluntary use by those other than IEEE and EIA members, whether the standard is to be used either domestically or internationally.

Standards and publications are approved by IEEE and EIA in accordance with the American National Standards Institute (ANSI) patent policy. By such action, IEEE and EIA do not assume any liability to any patent owner, nor do they assume any obligation whatever to parties adopting the Standard or Publication.

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. IEEE and EIA shall not be responsible for identifying all patents for which a license may be required by an IEEE and EIA standard or for conducting inquiries into the legal validity or scope of those patents that are brought to their 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.

Contents

Foreword to IEEE/EIA 12207.0-1996.....	v
--	---

ISO/IEC 12207 : 1995

Foreword	ix
Introduction.....	x
1. Scope.....	1
2. Normative references.....	2
3. Definitions	3
4. Application of this International Standard.....	6
5. Primary life cycle processes	9
5.1 Acquisition process.....	10
5.2 Supply process	13
5.3 Development process	16
5.4 Operation process.....	23
5.5 Maintenance process	24
6. Supporting processes.....	27
6.1 Documentation process.....	28
6.2 Configuration management process.....	29
6.3 Quality assurance process	31
6.4 Verification process.....	33
6.5 Validation process.....	36
6.6 Joint review process.....	38
6.7 Audit process.....	40
6.8 Problem resolution process.....	41
7. Organizational life cycle processes.....	42
7.1 Management process	43
7.2 Infrastructure process.....	45
7.3 Improvement process.....	46
7.4 Training process.....	47

Annexes

A—Tailoring process.....	48
B—Guidance on tailoring	49
C—Guidance on processes and organizations	53
D—Bibliography.....	57

IEEE/EIA 12207.0-1996

E—Basic concepts of ISO/IEC 12207 58

F—Compliance 64

G—Life cycle processes objectives 66

H—Life cycle data objectives 72

I—Relationships 74

J—Errata 75

Foreword to IEEE/EIA 12207.0-1996

1. The International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) published ISO/IEC 12207, *Information technology—Software life cycle processes*, in August, 1995. IEEE/EIA 12207.0 consists of the clarifications, additions, and changes accepted by the Institute of Electrical and Electronics Engineers (IEEE) and the Electronic Industries Alliance (EIA) as formulated by a joint project of the two organizations. IEEE/EIA 12207.0 contains concepts and guidelines to foster better understanding and application of the standard. Thus this standard provides industry a basis for software practices that would be usable for both national and international business. Readers are advised to refer to the information on Basic concepts (annex E), Compliance (annex F), Life cycle processes objectives (annex G), Life cycle data objectives (annex H), Relationships (annex I), and Errata (annex J) prior to implementing this standard. Annexes E through J have been added to IEEE/EIA 12207.0-1996 to enhance the reader's understanding of this standard.
2. IEEE/EIA 12207.0 may be used to
 - a. Acquire, supply, develop, operate, and maintain software.
 - b. Support the above functions in the form of quality assurance, configuration management, joint reviews, audits, verification, validation, problem resolution, and documentation.
 - c. Manage and improve the organization's processes and personnel.
 - d. Establish software management and engineering environments based upon the life cycle processes as adapted and tailored to serve business needs.
 - e. Foster improved understanding between customers and vendors and among the parties involved in the life cycle of a software product.
 - f. Facilitate world trade in software.
3. IEEE/EIA 12207 is packaged in three parts. The three parts are, briefly, as follows:

IEEE/EIA 12207.0, *Standard for Information Technology—Software life cycle processes*: Contains ISO/IEC 12207 in its original form and six additional annexes (E through J): Basic concepts; Compliance; Life cycle process objectives; Life cycle data objectives; Relationships; and Errata. A unique IEEE/EIA foreword is included.

IEEE/EIA P12207.1¹, *Guide for ISO/IEC 12207, Standard for Information Technology—Software life cycle processes—Life cycle data*: Provides additional guidance on recording life cycle data.

IEEE/EIA P12207.2, *Guide for ISO/IEC 12207, Standard for Information Technology—Software life cycle processes—Implementation considerations*: Provides additions, alternatives, and clarifications to the ISO/IEC 12207's life cycle processes as derived from U.S. practices.

¹ IEEE/EIA P12207.1 and P12207.2 are draft standards.

Participants in IEEE/EIA 12207.0

The unique front matter and annexes to IEEE/EIA 12207.0 were developed by the Joint Industrial Standard Working Group (JISWG) of IEEE and EIA, including the following persons:

Raghu Singh, *IEEE Co-chair*

Perry DeWeese, *EIA Co-chair*

Dorothy Kuckuck, *Project Editor*

Dennis Ahern
Chuck Baker
Barbara Bankeroff
Johnny Barrett
Ron Berlack
Ed Beverly
Barry Boehm
John Bolland
John Bowers
Max Brown
Don Calvert
Stuart Campbell
Quyen Cao
Bruce Capehart
Dennis Carter
Myra M. Chern
John P. Chihorek
Jack Cooper
Raymond Coyle
Paul Croll
Dario de Angelis
Chris Denham
Perry DeWeese
Robert Didrikson
Jim Dobbins
Merlin Dorfman
Cheryl Dorsey
Bernadette Downward
Peter Eirich
Bob Elston
Richard Evans
Dennis Faulkner
Marv Gechman
William Gess, Jr.

Marilyn Ginsberg-Finner
Lewis Gray
John Halase
John Hamlin
Rick Hefner
Robert Hegland
James H. Heil
Mark Henley
John Hoover
Helmut Hummel
Allan Jaworski
John Kerr
Lynne Ketchie
Larry Klos
Robert Knickerbocker
Steve Kopp
Richard Kreke
Dorothy Kuckuck
Jerome Lake
Doug Lange
Amy Laughlin
Milton Lavin
Jonathan Liles
Joan Lovelace
Marv Lubofsky
David Maibor
Ed Martin
Zygy Martynowicz
Richard McClellan
Judy McCloskey
Sandy McGill
Fred Mintz
Aretha Moore

James Moore
Jackie Morman
Gary Motchan
Bart Nigro
George Nowinski
Al Olsen
Myrna L. Olson
Sherry Paquin
Alex Polack
Ken Ptack
Ralph Randall
Paul Reindollar
Walter Richter
Bill Schunk
Adrienne Scott
Keith Shewbridge
Carl A. Singer
Terry Snyder
Reed Sorensen
Don Sova
John Stolzenthaler
Norma A. Stopyra
Richard F. Storch
Duane Stratton
Robert Tausworthe
Booker T. Thomas
Frederick Tilsworth
Leonard L. Tripp
Ann Turner
Howard Verne
Ronald L. Wade
David Waxman
Charles Wilson
Grady Wright

The following persons were on the IEEE balloting committee:

Syed Ali	Herbert Hecht	Alex Polack
Mikhail Auguston	William Hefley	Peter T. Poon
Robert E. Barry	Manfred Hein	Lawrence S. Przybylski
Leo Beltracchi	Mark Heinrich	Larry K. Reed
Mordechai Ben-Menachem	Mark Henley	Ann Reedy
H. Ronald Berlack	Umesh P. Hiriyannaiah	Annette D. Reilly
Christos Bezirtzoglou	John W. Horch	Dennis Rilling
William J. Boll	Jerry Huller	Patricia Rodriguez
Audrey C. Brewer	Peter L. Hung	Patti Rusher
Alan L. Bridges	Fabrizio Imelio	Andrew P. Sage
Kathleen L. Briggs	George Jackelen	Helmut Sandmayr
M. Scott Buck	Frank V. Jorgensen	Hans Schaefer
David W. Burnett	Vladan V. Jovanovic	Norman Schneidewind
Edward R. Byrne	William S. Junk	David J. Schultz
Stuart Ross Campbell	George X. Kambic	Robert W. Shillato
Leslie Chambers	Diana Kang	Katsutoshi Shintani
Keith Chan	Chris F. Kemerer	Carl A. Singer
Betty P. Chao	Judy Kerner	Raghu P. Singh
S. V. Chiyyarath	Robert J. Kierzyk	James M. Sivak
Antonio M. Cicu	Dwayne L. Knirk	Nancy M. Smith
Theo Clarke	Thomas M. Kurihara	Alfred R. Sorkowitz
Francois Coallier	John B. Lane	Donald W. Sova
Rosemary Coleman	J. Dennis Lawrence	Luca Spotorno
Virgil Lee Cooper	Randal Leavitt	Julia Stesney
Geoff Cozens	Michael Lines	Norma Stopyra
Gregory T. Daich	James J. Longbucco	Fred J. Strauss
Hillary Davidson	Dieter Look	Christine Brown Strysik
Bostjan K. Derganc	John Lord	Michael Surratt
Sanjay Dewal	M. Lubofsky	Toru Takeshita
Harpal Dhama	Austin J. Maher	Douglas H. Thiele
James Do	David Maibor	Booker Thomas
Audrey Dorofee	Harold Mains	Patricia Trellue
Carl Einar Dragstedt	Philip P. Mak	Leonard L. Tripp
Leo Egan	Gianluca Marcellino	T. H. Tse
Richard L. Evans	John R. Matras	Richard D. Tucker
William Eventoff	Tomoo Matsubara	Margaret C. Updike
Jonathan H. Fairclough	Mike McAndrew	Theodore J. Urbanowicz
John W. Fendrich	Patrick McCray	Glenn D. Venables
Jay Forster	Russell McDowell	David W. Vickers
Kirby Fortenberry	Sue McGrath	Udo Voges
Eva Freund	Jerome W. Mersky	Thomas E. Vollman
Adel N. Ghannam	Fatma Mili	Ronald L. Wade
Hiranmay Ghosh	Alan Miller	Dolores Wallace
Marilyn Ginsberg-Finner	Lisa Ming	John W. Walz
M. Joel Gittleman	Millard Allen Mobley	Charles J. Wertz
John Garth Glynn	Charles S. Mooney	Camille S. White-Partain
Julio Gonzalez Sanz	James W. Moore	Scott A. Whitmire
Donald Gotterbarn	Pavol Navrat	Paul A. T. Wolfgang
Lewis Gray	Dennis E. Nickle	Paul R. Work
Eric Grosse	Myrna L. Olson	Forrest D. Wright
Lawrence M. Gunther	Mike Ottewill	Weider D. Yu
Jon Hagar	Gerald L. Ourada	Janusz Zalewski
John Harauz	Lalit Mohan Patnaik	Zhi Ying Zhou
Rob Harker	Mark Paulk	Geraldine Zimmerman
Robert T. Harley	John G. Phippen	Peter F. Zoll

When the IEEE Standards Board approved IEEE/EIA 12207.0 on 10 December 1996, it had the following membership:

Donald C. Loughry, *Chair*

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

*Member Emeritus

Richard J. Holleman, *Vice Chair*
Andrew G. Salem, *Secretary*

E. G. "Al" Kiener
Joseph L. Koepfinger*
Stephen R. Lambert
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

Also included are the following nonvoting IEEE Standards Board liaisons:

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

Kristin M. Dittmann
IEEE Standards Project Editor

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 12207 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software engineering*.

Annex A forms an integral part of the International Standard. Annexes B and C are for information only.

Introduction

Software is an integral part of information technology and conventional systems, such as transportation, military, medical care, and finance. There is a proliferation of standards, procedures, methods, tools, and environments for developing and managing software. This proliferation has created difficulties in software management and engineering, especially in integrating products and services. The software discipline needs to migrate from this proliferation to a common framework that can be used by software practitioners to "speak the same language" to create and manage software. This International Standard provides such a common framework.

The framework covers the life cycle of software from conceptualization of ideas through retirement and consists of processes for acquiring and supplying software products and services. In addition, the framework provides for controlling and improving these processes.

The processes in this International Standard form a comprehensive set. An organization, depending on its purpose, can select an appropriate subset to fulfill that purpose. The International Standard is, therefore, designed to be tailored for an individual organization, project, or application. It is also designed to be used when software is a stand-alone entity, or an embedded or integral part of the total system.

Information technology — Software life cycle processes

1 Scope

1.1 Purpose

This International Standard establishes a common framework for software life cycle processes, with well-defined terminology, that can be referenced by the software industry. It contains processes, activities, and tasks that are to be applied during the acquisition of a system that contains software, a stand-alone software product, and software service and during the supply, development, operation, and maintenance of software products. Software includes the software portion of firmware.

This International Standard also provides a process that can be employed for defining, controlling, and improving software life cycle processes.

1.2 Field of application

This International Standard applies to the acquisition of systems and software products and services, to the supply, development, operation, and maintenance of software products, and to the software portion of firmware, whether performed internally or externally to an organization. Those aspects of system definition needed to provide the context for software products and services are included.

NOTE—The processes used during the software life cycle need to be compatible with the processes used during the system life cycle.

This International Standard is intended for use in a two-party situation and may be equally applied where the two parties are from the same organization. The situation may range from an informal agreement up to a legally binding contract. This International Standard may be used by a single party as self-imposed tasks.

This International Standard is not intended for off-the-shelf software products unless incorporated into a deliverable product.

This International Standard is written for acquirers of systems and software products and services and for suppliers, developers, operators, maintainers, managers, quality assurance managers, and users of software products.

1.3 Tailoring of the International Standard

This International Standard contains a set of processes, activities, and tasks designed to be tailored in respect of software projects. The tailoring process is deletion of non-applicable processes, activities, and tasks.

NOTE—Addition of unique or special processes, activities, and tasks may be provided in the contract.

1.4 Compliance

Compliance with this International Standard is defined as the performance of all the processes, activities, and tasks selected from this International Standard in the Tailoring Process (annex A) for the software project. The performance of a process or an activity is complete when all its required tasks are performed in accordance with the pre-established criteria and the requirements specified in the contract as applicable.

Any organization (for example, national, industrial association, company) imposing this International Standard, as a condition of trade, is responsible for specifying and making public the minimum set of required processes, activities, and tasks, which constitute suppliers' compliance with this International Standard.

1.5 Limitations

This International Standard describes the architecture of the software life cycle processes but does not specify the details of how to implement or perform the activities and tasks included in the processes.

This International Standard is not intended to prescribe the name, format, or explicit content of the documentation to be produced. This International Standard may require development of documents of similar class or type; various plans are an example. This International Standard, however, does not imply that such documents be developed or packaged separately or combined in some fashion. These decisions are left to the user of this International Standard.

This International Standard does not prescribe a specific life cycle model or software development method. The parties of this International Standard are responsible for selecting a life cycle model for the software project and mapping the processes, activities, and tasks in this International Standard onto that model. The parties are also responsible for selecting and applying the software development methods and for performing the activities and tasks suitable for the software project.

This International Standard is not intended to be in conflict with any organization's policies, standards or procedures that are already in place. However, any conflict needs to be resolved and any overriding conditions and situations need to be cited in writing as exceptions to the application of this International Standard.

Throughout this International Standard, "shall" is used to express a provision that is binding between two or more parties, "will" to express a declaration of purpose or intent by one party, "should" to express a recommendation among other possibilities, and "may" to indicate a course of action permissible within the limits of this International Standard.

In this International Standard, there are a number of lists for tasks; none of these is presumed to be exhaustive—they are intended as examples.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/AFNOR: 1989, *Dictionary of computer science*.

ISO 2382-1: 1993, *Information technology—Vocabulary—Part 1: Fundamental terms*.

ISO/IEC 2382-20: 1990, *Information technology—Vocabulary; Part 20: System development*.

ISO 8402: 1994, *Quality management and quality assurance—Vocabulary*.

ISO 9001: 1994, *Quality systems—Models for quality assurance in design, development, production, installation and servicing*.

ISO/IEC 9126: 1991, *Information technology—Software product evaluation—Quality characteristics and guidelines for their use*.

3 Definitions

For the purposes of this International Standard, the definitions given in ISO 8402, ISO/IEC 2382-1 and ISO/IEC 2382-20 apply, together with the following definitions.

NOTE—A product may be interpreted as a part of a system as applicable.

3.1 Acquirer: An organization that acquires or procures a system, software product or software service from a supplier.

NOTE—The acquirer could be one of the following: buyer, customer, owner, user, purchaser.

3.2 Acquisition: The process of obtaining a system, software product or software service.

3.3 Agreement: The definition of terms and conditions under which a working relationship will be conducted.

3.4 Audit: Conducted by an authorized person for the purpose of providing an independent assessment of software products and processes in order to assess compliance with requirements.

3.5 Baseline: A formally approved version of a configuration item, regardless of media, formally designated and fixed at a specific time during the configuration item's life cycle.

3.6 Configuration item: An entity within a configuration that satisfies an end use function and that can be uniquely identified at a given reference point.

3.7 Contract: A binding agreement between two parties, especially enforceable by law, or a similar internal agreement wholly within an organization, for the supply of software service or for the supply, development, production, operation, or maintenance of a software product.

3.8 Developer: An organization that performs development activities (including requirements analysis, design, testing through acceptance) during the software life cycle process.

3.9 Evaluation: A systematic determination of the extent to which an entity meets its specified criteria.

3.10 Firmware: The combination of a hardware device and computer instructions or computer data that reside as read-only software on the hardware device. The software cannot be readily modified under program control.

3.11 Life cycle model: A framework containing the processes, activities, and tasks involved in the development, operation, and maintenance of a software product, spanning the life of the system from the definition of its requirements to the termination of its use.

3.12 Maintainer: An organization that performs maintenance activities.

3.13 Monitoring: An examination of the status of the activities of a supplier and of their results by the acquirer or a third party.

3.14 Non-deliverable item: Hardware or software product that is not required to be delivered under the contract but may be employed in the development of a software product.

3.15 Off-the-shelf product: Product that is already developed and available, usable either "as is" or with modification.

3.16 Operator: An organization that operates the system.

3.17 Process: A set of interrelated activities, which transform inputs into outputs.

NOTE—The term "activities" covers use of resources.

3.18 Qualification: The process of demonstrating whether an entity is capable of fulfilling specified requirements. [See ISO 8402: 1994, 2.13.]

3.19 Qualification requirement: A set of criteria or conditions that have to be met in order to qualify a software product as complying with its specifications and being ready for use in its target environment.

3.20 Qualification testing: Testing, conducted by the developer and witnessed by the acquirer (as appropriate), to demonstrate that the software product meets its specifications and is ready for use in its target environment.

3.21 Quality assurance: All the planned and systematic activities implemented within the quality system, and demonstrated as needed, to provide adequate confidence that an entity will fulfill requirements for quality.

NOTES

1 There are both internal and external purposes for quality assurance:

- a) Internal quality assurance: within an organization, quality assurance provides confidence to management;
- b) External quality assurance: in contractual situations, quality assurance provides confidence to the customer or others.

2 Some quality control and quality assurance actions are interrelated.

3 Unless requirements for quality fully reflect the needs of the user, quality assurance may not provide adequate confidence.

[ISO 8402: 1994, 3.5]

3.22 Release: A particular version of a configuration item that is made available for a specific purpose (for example, test release).

3.23 Request for proposal [tender]: A document used by the acquirer as the means to announce its intention to potential bidders to acquire a specified system, software product or software service.

3.24 Retirement: Withdrawal of active support by the operation and maintenance organization, partial or total replacement by a new system, or installation of an upgraded system.

3.25 Security: The protection of information and data so that unauthorized persons or systems cannot read or modify them and authorized persons or systems are not denied access to them.

3.26 Software product: The set of computer programs, procedures, and possibly associated documentation and data.

3.27 Software service: Performance of activities, work, or duties connected with a software product, such as its development, maintenance, and operation.

3.28 Software unit: A separately compilable piece of code.

3.29 Statement of work: A document used by the acquirer as the means to describe and specify the tasks to be performed under the contract.

3.30 Supplier: An organization that enters into a contract with the acquirer for the supply of a system, software product or software service under the terms of the contract.

NOTES

1 The term "supplier" is synonymous with contractor, producer, seller, or vendor.

2 The acquirer may designate a part of its organization as supplier.

3.31 System: An integrated composite that consists of one or more of the processes, hardware, software, facilities and people, that provides a capability to satisfy a stated need or objective.

3.32 Test coverage: The extent to which the test cases test the requirements for the system or software product.

3.33 Testability: The extent to which an objective and feasible test can be designed to determine whether a requirement is met.

3.34 User: An individual or organization that uses the operational system to perform a specific function.

NOTE—The user may perform other roles such as acquirer, developer, or maintainer.

3.35 Validation: Confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled.

NOTES

1 In design and development, validation concerns the process of examining a product to determine conformity with user needs.

2 Validation is normally performed on the final product under defined operating conditions. It may be necessary in earlier stages.

3 "Validated" is used to designate the corresponding status.

4 Multiple validations may be carried out if there are different intended uses.

[ISO 8402: 1994, 2.18]

3.36 Verification: Confirmation by examination and provision of objective evidence that specified requirements have been fulfilled.

NOTES

1 In design and development, verification concerns the process of examining the result of a given activity to determine conformity with the stated requirement for that activity.

2 "Verified" is used to designate the corresponding status.

[ISO 8402: 1994, 2.17]

3.37 Version: An identified instance of an item.

NOTE—Modification to a version of a software product, resulting in a new version, requires configuration management action.

4 Application of this International Standard

This clause presents the software life cycle processes that can be employed to acquire, supply, develop, operate, and maintain software products. The objective is to provide a road map for the users of this International Standard so that they can orient themselves in it and apply it judiciously.

4.1 Organization of this International Standard

4.1.1 Life cycle processes

This International Standard groups the activities that may be performed during the life cycle of software into five primary processes, eight supporting processes, and four organizational processes. Each life cycle process is divided into a set of activities; each activity is further divided into a set of tasks. Subclause numbering a.b denotes a process, a.b.c an activity, and a.b.c.d a task. These life cycle processes are introduced below and depicted in figure 1.

4.1.1.1 Primary processes

The primary processes (clause 5) consist of five processes that serve primary parties during the life cycle of software. A primary party is one that initiates or performs the development, operation, or maintenance of software products. These primary parties are the acquirer, the supplier, the developer, the operator, and the maintainer of software products. The primary processes are:

- 1) *Acquisition process* (subclause 5.1). Defines the activities of the acquirer, the organization that acquires a system, software product or software service.
- 2) *Supply process* (subclause 5.2). Defines the activities of the supplier, the organization that provides the system, software product or software service to the acquirer.
- 3) *Development process* (subclause 5.3). Defines the activities of the developer, the organization that defines and develops the software product.
- 4) *Operation process* (subclause 5.4). Defines the activities of the operator, the organization that provides the service of operating a computer system in its live environment for its users.
- 5) *Maintenance process* (subclause 5.5). Defines the activities of the maintainer, the organization that provides the service of maintaining the software product; that is, managing modifications to the software product to keep it current and in operational fitness. This process includes the migration and retirement of the software product.

4.1.1.2 Supporting life cycle processes

The supporting life cycle processes (clause 6) consist of eight processes. A supporting process supports another process as an integral part with a distinct purpose and contributes to the success and quality of the software project. A supporting process is employed and executed, as needed, by another process. The supporting processes are:

- 1) *Documentation process* (subclause 6.1). Defines the activities for recording the information produced by a life cycle process.
- 2) *Configuration management process* (subclause 6.2). Defines the configuration management activities.
- 3) *Quality assurance process* (subclause 6.3). Defines the activities for objectively assuring that the software products and processes are in conformance with their specified requirements and adhere to their established plans. Joint Reviews, Audits, Verification, and Validation may be used as techniques of Quality Assurance.

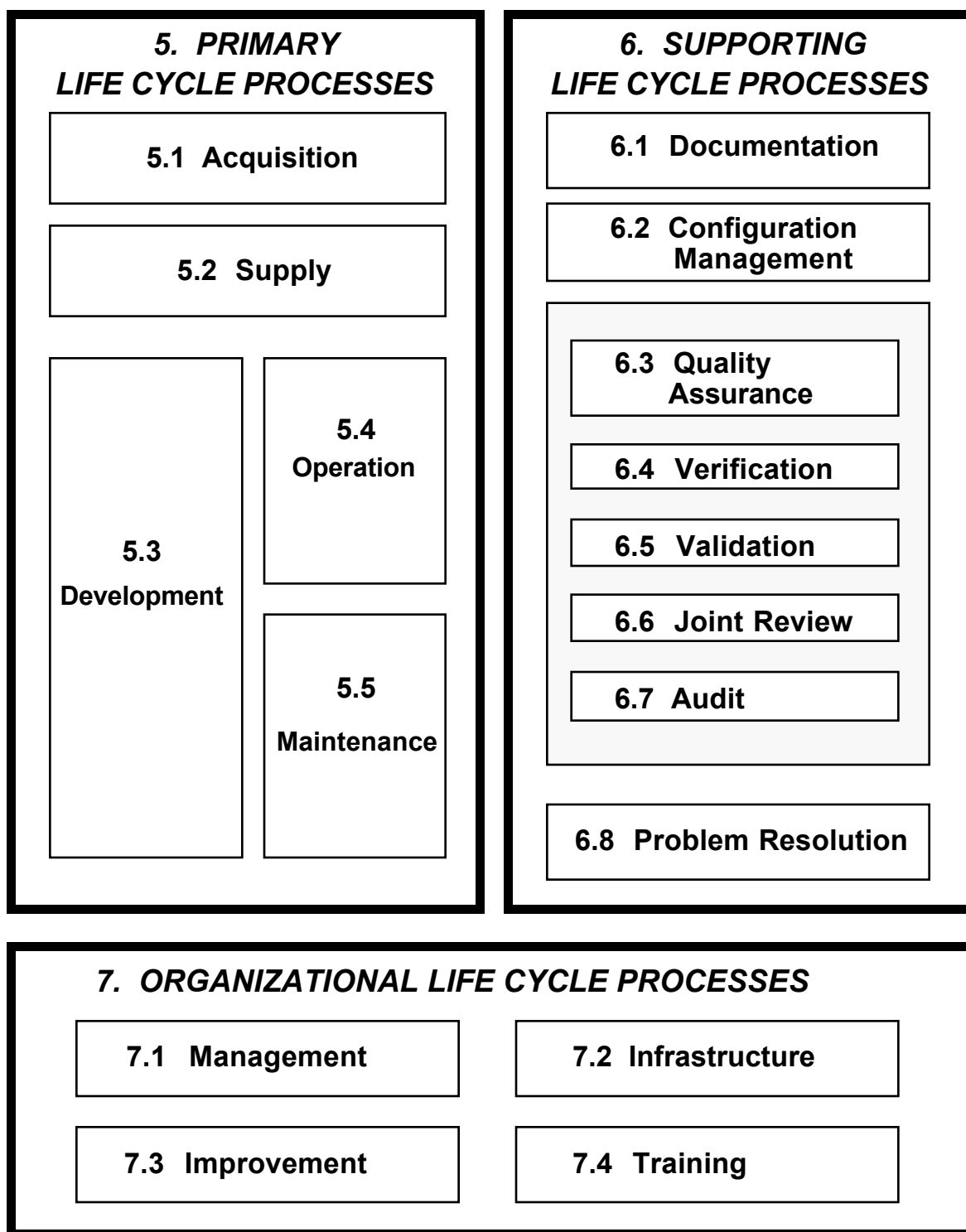


Figure 1. Structure of this International Standard

- 4) *Verification process* (subclause 6.4). Defines the activities (for the acquirer, the supplier, or an independent party) for verifying the software products and services in varying depth depending on the software project.
- 5) *Validation process* (subclause 6.5). Defines the activities (for the acquirer, the supplier, or an independent party) for validating the software products of the software project.
- 6) *Joint review process* (subclause 6.6). Defines the activities for evaluating the status and products of an activity. This process may be employed by any two parties, where one party (reviewing party) reviews another party (reviewed party) in a joint forum.
- 7) *Audit process* (subclause 6.7). Defines the activities for determining compliance with the requirements, plans, and contract. This process may be employed by any two parties, where one party (auditing party) audits the software products or activities of another party (audited party).
- 8) *Problem resolution process* (subclause 6.8). Defines a process for analyzing and removing the problems (including nonconformances), whatever their nature or source, that are discovered during the execution of development, operation, maintenance, or other processes.

4.1.1.3 Organizational life cycle processes

The organizational life cycle processes (clause 7) consist of four processes. They are employed by an organization to establish and implement an underlying structure made up of associated life cycle processes and personnel and continuously improve the structure and processes. They are typically employed outside the realm of specific projects and contracts; however, lessons from such projects and contracts contribute to the improvement of the organization. The organizational processes are:

- 1) *Management process* (subclause 7.1). Defines the basic activities of the management, including project management, related to the execution of a life cycle process.
- 2) *Infrastructure process* (subclause 7.2). Defines the basic activities for establishing the underlying structure of a life cycle process.
- 3) *Improvement process* (subclause 7.3). Defines the basic activities that an organization (that is, acquirer, supplier, developer, operator, maintainer, or the manager of another process) performs for establishing, measuring, controlling, and improving its life cycle process.
- 4) *Training process* (subclause 7.4). Defines the activities for providing adequately trained personnel.

4.1.2 Tailoring process. Annex A, which is normative, defines the basic activities needed to perform tailoring of this International Standard. Annex B contains a brief guidance on tailoring the requirements of this International Standard; it lists the key factors upon which tailoring decisions may be made.

4.1.3 Relationship between the processes and organizations

This International Standard contains various processes that are applied throughout the life cycle of software by various organizations depending on their needs and goals. For understandability, Annex C presents the relationships between the life cycle processes and related parties.

5 Primary life cycle processes

This clause defines the following primary life cycle processes:

- 1) Acquisition process;
- 2) Supply process;
- 3) Development process;
- 4) Operation process;
- 5) Maintenance process.

The activities and tasks in a primary process are the responsibility of the organization initiating and performing that process. This organization ensures that the process is in existence and functional.

5.1 Acquisition process

The Acquisition Process contains the activities and tasks of the acquirer. The process begins with the definition of the need to acquire a system, software product or software service. The process continues with the preparation and issue of a request for proposal, selection of a supplier, and management of the acquisition process through to the acceptance of the system, software product or software service.

The individual organization having the need may be called the owner. The owner may contract any or all of the acquisition activities to an agent who will in turn conduct these activities according to the Acquisition Process. The acquirer in this subclause may be the owner or the agent.

The acquirer manages the Acquisition Process at the project level following the Management Process (7.1), which is instantiated in this process; establishes an infrastructure under the process following the Infrastructure Process (7.2); tailors the process for the project following the Tailoring Process (annex A); and manages the process at the organizational level following the Improvement Process (7.3) and the Training Process (7.4).

List of activities: This process consists of the following activities:

- 1) Initiation;
- 2) Request-for-Proposal [-tender] preparation;
- 3) Contract preparation and update;
- 4) Supplier monitoring;
- 5) Acceptance and completion.

5.1.1 Initiation. This activity consists of the following tasks:

5.1.1.1 The acquirer begins the acquisition process by describing a concept or a need to acquire, develop, or enhance a system, software product or software service.

5.1.1.2 The acquirer will define and analyze the system requirements. The system requirements should include business, organizational and user as well as safety, security, and other criticality requirements along with related design, testing, and compliance standards and procedures.

5.1.1.3 If the acquirer retains a supplier to perform system requirements analysis, the acquirer will approve the analyzed requirements.

5.1.1.4 The acquirer may perform the definition and analysis of software requirements by itself or may retain a supplier to perform this task.

5.1.1.5 The Development Process (5.3) should be used to perform the tasks in 5.1.1.2 and 5.1.1.4.

5.1.1.6 The acquirer will consider options for acquisition against analysis of appropriate criteria to include risk, cost and benefits for each option. Options include:

- a) Purchase an off-the-shelf software product that satisfies the requirements.
- b) Develop the software product or obtain the software service internally.
- c) Develop the software product or obtain the software service through contract.
- d) A combination of a, b, and c above.
- e) Enhance an existing software product or service.

5.1.1.7 When an off-the-shelf software product is to be acquired, the acquirer will ensure the following conditions are satisfied:

- a) The requirements for the software product are satisfied.
- b) The documentation is available.
- c) Proprietary, usage, ownership, warranty and licensing rights are satisfied.
- d) Future support for the software product is planned.

5.1.1.8 The acquirer should prepare, document and execute an acquisition plan. The plan should contain the following:

- a) Requirements for the system;
- b) Planned employment of the system;
- c) Type of contract to be employed;
- d) Responsibilities of the organizations involved;
- e) Support concept to be used;
- f) Risks considered as well as methods to manage the risks.

5.1.1.9 The acquirer should define and document the acceptance strategy and conditions (criteria).

5.1.2 Request-for-proposal [tender] preparation. This activity consists of the following tasks:

5.1.2.1 The acquirer should document the acquisition requirements (e.g., request for proposal), the content of which depends upon the acquisition option selected in 5.1.1.6. The acquisition documentation should include, as appropriate:

- a) System requirements;
- b) Scope statement;
- c) Instructions for bidders;
- d) List of software products;
- e) Terms and conditions;
- f) Control of subcontracts;
- g) Technical constraints (e.g., target environment).

5.1.2.2 The acquirer should determine which processes, activities, and tasks of this International Standard are appropriate for the project and should tailor them accordingly. Especially, the acquirer should specify the applicable supporting processes (clause 6) and their performing organizations, including responsibilities (if other than supplier), so that the suppliers may, in their proposals, define the approach to each of the specified supporting processes. The acquirer will define the scope of those tasks that reference the contract.

5.1.2.3 The acquisition documentation will also define the contract milestones at which the supplier's progress will be reviewed and audited as part of monitoring the acquisition (see 6.6 and 6.7).

5.1.2.4 The acquisition requirements should be given to the organization selected for performing the acquisition activities.

5.1.3 Contract preparation and update. This activity consists of the following tasks:

5.1.3.1 The acquirer should establish a procedure for supplier selection including proposal evaluation criteria and requirements compliance weighting.

5.1.3.2 The acquirer should select a supplier based upon the evaluation of the suppliers' proposals, capabilities, and other factors that need to be considered.

5.1.3.3 The acquirer may involve other parties, including potential suppliers, before contract award, in tailoring this International Standard for the project. However, the acquirer will make the final decision on the tailoring. The acquirer will include or reference the tailored International Standard in the contract.

5.1.3.4 The acquirer will then prepare and negotiate a contract with the supplier, that addresses the acquisition requirements, including the cost and schedule, of the software product or service to be delivered. The contract will address proprietary, usage, ownership, warranty and licensing rights associated with the reusable off-the-shelf software products.

5.1.3.5 Once the contract is underway, the acquirer will control changes to the contract through negotiation with the supplier as part of a change control mechanism. Changes to the contract shall be investigated for impact on project plans, costs, benefits, quality, and schedule.

NOTE—The acquirer determines whether the term "contract" or "agreement" is to be used in the application of this International Standard.

5.1.4 Supplier monitoring. This activity consists of the following tasks:

5.1.4.1 The acquirer will monitor the supplier's activities in accordance with the Joint Review Process (6.6) and the Audit Process (6.7). The acquirer should supplement the monitoring with the Verification Process (6.4) and the Validation Process (6.5) as needed.

5.1.4.2 The acquirer will cooperate with the supplier to provide all necessary information in a timely manner and resolve all pending items.

5.1.5 Acceptance and completion. This activity consists of the following tasks:

5.1.5.1 The acquirer should prepare for acceptance based on the defined acceptance strategy and criteria. The preparation of test cases, test data, test procedures, and test environment should be included. The extent of supplier involvement should be defined.

5.1.5.2 The acquirer will conduct acceptance review and acceptance testing of the deliverable software product or service and will accept it from the supplier when all acceptance conditions are satisfied. The acceptance procedure should comply with the provisions of 5.1.1.9.

5.1.5.3 After acceptance, the acquirer should take the responsibility for the configuration management of the delivered software product (see 6.2).

NOTE—The acquirer may install the software product or perform the software service in accordance with instructions defined by the supplier.

5.2 Supply process

The Supply Process contains the activities and tasks of the supplier. The process may be initiated either by a decision to prepare a proposal to answer an acquirer's request for proposal or by signing and entering into a contract with the acquirer to provide the system, software product or software service. The process continues with the determination of procedures and resources needed to manage and assure the project, including development of project plans and execution of the plans through delivery of the system, software product or software service to the acquirer.

The supplier manages the Supply Process at the project level following the Management Process (7.1), which is instantiated in this process; establishes an infrastructure under the process following the Infrastructure Process (7.2); tailors the process for the project following the Tailoring Process (annex A); and manages the process at the organizational level following the Improvement Process (7.3) and the Training Process (7.4).

List of activities: This process consists of the following activities:

- 1) Initiation;
- 2) Preparation of response;
- 3) Contract;
- 4) Planning;
- 5) Execution and control;
- 6) Review and evaluation;
- 7) Delivery and completion.

5.2.1 Initiation. This activity consists of the following tasks:

5.2.1.1 The supplier conducts a review of requirements in the request for proposal taking into account organizational policies and other regulations.

5.2.1.2 The supplier should make a decision to bid or accept the contract.

5.2.2 Preparation of response. This activity consists of the following task:

5.2.2.1 The supplier should define and prepare a proposal in response to the request for proposal, including its recommended tailoring of this International Standard.

5.2.3 Contract. This activity consists of the following tasks:

5.2.3.1 The supplier shall negotiate and enter into a contract with the acquirer organization to provide the software product or service.

5.2.3.2 The supplier may request modification to the contract as part of the change control mechanism.

5.2.4 Planning. This activity consists of the following tasks:

5.2.4.1 The supplier shall conduct a review of the acquisition requirements to define the framework for managing and assuring the project and for assuring the quality of the deliverable software product or service.

5.2.4.2 If not stipulated in the contract, the supplier shall define or select a software life cycle model appropriate to the scope, magnitude, and complexity of the project. The processes, activities, and tasks of this International Standard shall be selected and mapped onto the life cycle model.

5.2.4.3 The supplier shall establish requirements for the plans for managing and assuring the project and for assuring the quality of the deliverable software product or service. Requirements for the plans should include resource needs and acquirer involvement.

5.2.4.4 Once the planning requirements are established, the supplier shall consider the options for developing the software product or providing the software service, against an analysis of risks associated with each option. Options include:

- a) Develop the software product or provide the software service using internal resources.
- b) Develop the software product or provide the software service by subcontracting.

- c) Obtain off-the-shelf software products from internal or external sources.
- d) A combination of a, b, and c above.

5.2.4.5 The supplier shall develop and document project management plan(s) based upon the planning requirements and options selected in 5.2.4.4. Items to be considered in the plan include but are not limited to the following:

- a) Project organizational structure and authority and responsibility of each organizational unit, including external organizations;
- b) Engineering environment (for development, operation, or maintenance, as applicable), including test environment, library, equipment, facilities, standards, procedures, and tools;
- c) Work breakdown structure of the life cycle processes and activities, including the software products, software services and non-deliverable items, to be performed together with budgets, staffing, physical resources, software size, and schedules associated with the tasks;
- d) Management of the quality characteristics of the software products or services. Separate plans for quality may be developed.
- e) Management of the safety, security, and other critical requirements of the software products or services. Separate plans for safety and security may be developed.
- f) Subcontractor management, including subcontractor selection and involvement between the subcontractor and the acquirer, if any;
- g) Quality assurance (see 6.3);
- h) Verification (see 6.4) and validation (see 6.5); including the approach for interfacing with the verification and validation agent, if specified;
- i) Acquirer involvement; that is, by such means as joint reviews (see 6.6), audits (see 6.7), informal meetings, reporting, modification and change; implementation, approval, acceptance, and access to facilities;
- j) User involvement; by such means as requirements setting exercises, prototype demonstrations and evaluations;
- k) Risk management; that is management of the areas of the project that involve potential technical, cost, and schedule risks;
- l) Security policy; that is, the rules for need-to-know and access-to-information at each project organization level;
- m) Approval required by such means as regulations, required certifications, proprietary, usage, ownership, warranty and licensing rights;
- n) Means for scheduling, tracking, and reporting;
- o) Training of personnel (see 7.4).

5.2.5 Execution and control. This activity consists of the following tasks:

5.2.5.1 The supplier shall implement and execute the project management plan(s) developed in 5.2.4.

5.2.5.2 The supplier shall:

- a) Develop the software product in accordance with Development Process (5.3).
- b) Operate the software product in accordance with Operation Process (5.4).
- c) Maintain the software product in accordance with Maintenance Process (5.5).

5.2.5.3 The supplier shall monitor and control the progress and the quality of the software products or services of the project throughout the contracted life cycle. This shall be an ongoing, iterative task, which shall provide for:

- a) Monitoring progress of technical performance, costs, and schedules and reporting of project status;
- b) Problem identification, recording, analysis, and resolution.

5.2.5.4 The supplier shall manage and control the subcontractors in accordance with the Acquisition Process (5.1). The supplier shall pass down all contractual requirements necessary to ensure that the software product or service delivered to the acquirer is developed or performed in accordance with the prime-contract requirements.

5.2.5.5 The supplier shall interface with the independent verification, validation, or test agent as specified in the contract and project plans.

5.2.5.6 The supplier shall interface with other parties as specified in the contract and project plans.

5.2.6 Review and evaluation. This activity consists of the following tasks:

5.2.6.1 The supplier should coordinate contract review activities, interfaces, and communication with the acquirer's organization.

5.2.6.2 The supplier shall conduct or support the informal meetings, acceptance review, acceptance testing, joint reviews, and audits with the acquirer as specified in the contract and project plans. The joint reviews shall be conducted in accordance with 6.6, audits in accordance with 6.7.

5.2.6.3 The supplier shall perform verification and validation in accordance with 6.4 and 6.5 respectively to demonstrate that the software products or services and processes fully satisfy their respective requirements.

5.2.6.4 The supplier shall make available to the acquirer the reports of evaluation, reviews, audits, testing, and problem resolutions as specified in the contract.

5.2.6.5 The supplier shall provide the acquirer access to the supplier's and subcontractors' facilities for review of software products or services as specified in the contract and project plans.

5.2.6.6 The supplier shall perform quality assurance activities in accordance with 6.3.

5.2.7 Delivery and completion. This activity consists of the following tasks:

5.2.7.1 The supplier shall deliver the software product or service as specified in the contract.

5.2.7.2 The supplier shall provide assistance to the acquirer in support of the delivered software product or service as specified in the contract.