

IEEE Recommended Practice for Software Acquisition

Sponsor

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

Approved December 2, 1993

IEEE Standards Board

Abstract: A set of useful quality practices that can be selected and applied during one or more steps in a software acquisition process is described. This recommended practice can be applied to software that runs on any computer system regardless of the size, complexity, or criticality of the software, but is more suited for use on modified-off-the-shelf software and fully developed software.

Keywords: acquirer, modified-off-the-shelf software, software acquisition life cycle, software acquisition process, supplier

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

Copyright © 1994 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1994. Printed in the United States of America.

ISBN 1-55937-385-7

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 may involve the use of patented technology. Their approval by the Institute of Electrical and Electronics Engineers does not mean that using such technology for the purpose of conforming to such standards is authorized by the patent owner. It is the obligation of the user of such technology to obtain all necessary permissions.

Introduction

(This introduction is not a part of IEEE Std 1062-1993, IEEE Recommended Practice for Software Acquisition.)

This introduction provides some background on the rationale used to develop this recommended practice. This information is meant to aid in the understanding and usage of this recommended practice.

This recommended practice describes the management and execution of software acquisition activities. It is intended for:

- a) Individuals or organizations who use software and acquire that software from suppliers
- b) Individuals or organizations who acquire software from a developer for resale to other individuals or organizations
- c) Individuals or organizations who influence how software is acquired from suppliers
- d) Suppliers interested in providing high-quality software to acquirers

This recommended practice is designed to help organizations and individuals:

- a) Incorporate quality considerations during the definition, evaluation, selection, and acceptance of supplier software for operational use
- b) Determine how supplier software should be evaluated, tested, and accepted for delivery to end users

This recommended practice is intended to satisfy the following objectives:

- a) Promote consistency within organizations in acquiring third-party software from software suppliers
- b) Provide useful practices on including quality considerations during acquisition planning
- c) Provide useful practices on evaluating and qualifying supplier capabilities to meet user requirements
- d) Provide useful practices on evaluating and qualifying supplier software
- e) Assist individuals or organizations judging the quality of supplier software for referral to end users

Participants

At the time this recommended practice was completed, the Software Acquisition Working Group had the following membership:

Phillip C. Marriott, *Chair*

Karen Allison
Ellen M. Augustine
Mordechai Ben-Menachem
David L. Boudreau
Rick Burgess
Edward R. Byrne
Liliane Choney
Christopher Cooke
Rena Crabill
Patricia M. Daggett
Linda Uca Domiani
Dennis J. Eaton
Leo G. Egan
Jane Frederick
Thomas Gray

Flo Harteloo, *Vice Chair*

Nicholsa P. Ginex
Paul Haller
Frank Henninger
Lloyd Johnson
Anna Johnston
Miles Kehoe
Bob Kessler
C. L. (Kelly) Kjelstrom
Larry King
Thomas M. Kurihara
Rebecca Ann Lamb
A. Lip Lim
F. C. Lim
Ben Livson
John J. McKissick, Jr.

Louis R. Mills
Rafael E. Padilla
Duane K. Palmiter
John Passafiume
Sandy Raddue
Robert H. Randolph
Horst P. Richter
David P. Schwartz
Mark Shelton
David M. Siefert
Paulette Spink
Kenneth Thoreson
William Timby
George W. Trever
William S. Turner, III

The following persons were on the balloting committee:

W. Arfvidson
M. Azuma
B. Banerjee
M. Ben-Menachem
H. R. Berlack
W. J. Boll, Jr.
K. L. Briggs
B. Brocka
F. Buckley
F. Coallier
P. W. Daggett
T. Daughtrey
L. G. Egan
C. L. Evans
J. W. Fendrich
R. G. Fordham

Y. Gershkovich
J. Gonzalez-Sanz
D. A. Gustafson
W. Hefley
C. P. Hollocker
J. W. Horch
D. Johnson
P. Klopfenstein
T. M. Kurihara
R. A. Lamb
J. B. Lane
F. C. Lim
B. Lindberg
B. Livson
J. Maayan
H. Mains

P. C. Marriott
R. Martin
I. Mazza
D. E. Nickle
J. G. Phippen
J. D. Pope
S. R. Schach
H. Schaefer
W. A. Schnoege
G. D. Schumacher
D. M. Siefert
L. L. Tripp
D. Wallace
A. Wilson
P. A. Wolfgang
N. C. Yopconka

When the IEEE Standards Board approved this standard on December 2, 1993, it had the following membership:

Wallace S. Read, *Chair*

Donald C. Loughry, *Vice Chair*

Andrew G. Salem, *Secretary*

Gilles A. Baril
José A. Berrios de la Paz
Clyde R. Camp
Donald C. Fleckenstein
Jay Forster*
David F. Franklin
Ramiro Garcia
Donald N. Heirman

Jim Isaak
Ben C. Johnson
Walter J. Karplus
Lorraine C. Kevra
E. G. "Al" Kiener
Ivor N. Knight
Joseph L. Koepfinger*
D. N. "Jim" Logothetis

Don T. Michael*
Marco W. Migliaro
L. John Rankine
Arthur K. Reilly
Ronald H. Reimer
Gary S. Robinson
Leonard L. Tripp
Donald W. Zipse

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal
James Beall
Richard B. Engelman
David E. Soffrin
Stanley I. Warshaw

Rachel A. Meisel
IEEE Standards Project Editor

Contents

CLAUSE	PAGE
1. Overview	1
1.1 Scope.....	1
1.2 Terminology.....	2
2. References.....	3
3. Definitions.....	5
4. Introducing the software acquisition process.....	6
4.1 Software acquisition life cycle.....	6
4.2 Nine steps in acquiring quality software.....	7
5. Software acquisition process.....	10
5.1 Planning organizational strategy.....	10
5.2 Implementing organization's process	10
5.3 Defining the software requirements.....	11
5.4 Identifying potential suppliers	12
5.5 Preparing contract requirements	14
5.6 Evaluating proposals and selecting supplier	15
5.7 Managing for supplier performance.....	16
5.8 Accepting the software	17
5.9 Using the software	18
6. Summary	19
Annex A Checklists to assist organizations in establishing their own software acquisition process.....	20

IEEE Recommended Practice for Software Acquisition

1. Overview

This recommended practice is divided into six clauses. Clause 1 provides the scope of this recommended practice. Clause 2 lists references to other standards that are useful in applying this recommended practice. Clause 3 provides definitions that are either not found in other standards, or have been modified for use with this recommended practice. Clause 4 establishes the nine steps involved in a software acquisition process, relates each of these steps to a major acquisition phase, and identifies the key inputs and outputs of each step. Clause 5 describes the nine steps in the software acquisition process and the related quality practices that apply to acquiring software. In order to be in compliance with this recommended practice, clause 5 must be adhered to. Clause 6 summarizes the successful way to acquire high-quality products and services from software suppliers.

This recommended practice also contains an annex. Annex A provides a set of checklists that individuals or organizations may elect to adapt to their specific needs.

1.1 Scope

This is a recommended practice for performing software acquisitions. It describes a set of useful quality practices that can be selected and applied during one or more steps in a software acquisition process.

In this recommended practice, software products have been classified according to the degree to which the acquirer may specify the features of the software. They are: commercial-off-the-shelf (COTS), modified-off-the-shelf (MOTS), and fully developed item.

Commercial-off-the-shelf (COTS) software is stable and is normally well-defined in terms of documentation and known capabilities and limitations. It usually comes with “how to operate” documentation. COTS software is defined by a market-driven need. It is commercially available and its fitness for use has been demonstrated by a broad spectrum of commercial users. Also, the COTS software supplier does not advertise any willingness to modify the software for a specific customer.

Modified-off-the-shelf (MOTS) software is similar to COTS software; however, the MOTS software does advertise service to tailor the software to acquirer-specific requirements.

Fully developed software will often be unique for a specific application and will be produced on a one-of-a-kind or low-volume basis. The software typically will have the potential for future modification by the acquirer to meet changing needs. As a result, most of the documentation will be special to the project (with

the exception of the supplier's standard documentation for the operating system, any standard application packages, and programming languages).

This recommended practice can be applied to software that runs on any computer system regardless of the size, complexity, or criticality of the software. However, this recommended practice is more suited for use on modified-off-the-shelf software and fully developed software. Each organization using this recommended practice will need to identify the classes of software to which this recommended practice applies and the specific quality characteristics and activities that need to be included within the acquisition process.

1.2 Terminology

The words *shall* and *must* identify the mandatory (essential) material with this recommended practice. The words *should* and *may* identify optional (conditional) material. The terminology in this recommended practice is based on IEEE Std 610.12-1990.¹ New terms and modified definitions as applied in this recommended practice can be found in clause 3.

¹Information on references can be found in clause 2.

2. References

The following standards are directly referenced in this recommended practice. Table 1 provides a cross-reference of standards that address topics related to software acquisition. These standards are binding to the extent referenced within the text of this recommended practice and are referenced to avoid duplication of requirements.

IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology (ANSI).²

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

IEEE Std 829-1983 (Reaff 1991), IEEE Standard for Software Test Documentation (ANSI).

IEEE Std 830-1993, IEEE Guide to Software Requirements Specifications.³

IEEE Std 1012-1986, IEEE Standard for Software Verification and Validation Plans (ANSI).

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

IEEE Std 1028-1988, IEEE Standard for Software Reviews and Audits (ANSI).

IEEE Std 1042-1987, IEEE Guide for Software Configuration Management (ANSI).

IEEE Std 1045-1992, IEEE Standard for Software Productivity Metrics (ANSI).

IEEE Std 1058-1988, IEEE Standard for Software Project Management Plans.

IEEE Std 1061-1992, IEEE Standard for Software Quality Metrics Methodology (ANSI).

IEEE Std 1063-1987, IEEE Standard for Software User Documentation (ANSI).

IEEE Std 1074-1991, IEEE Standard for Developing Software Life Cycle Processes (ANSI).

IEEE Std 1209-1992, IEEE Recommend Practice for the Evaluation and Selection of CASE Tools (ANSI).

IEEE Std 1219-1992, IEEE Standard for Software Maintenance (ANSI).

ISO/IEC 9126 : 1991, Information technology—Software product evaluation—Quality characteristics and guidelines for their use.⁴

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

³As this standard goes to press, IEEE Std 830-1993 is not yet published. It is, however, available in manuscript form from the IEEE Standards Department, (908) 562-3800. Anticipated publication date is April 1994, at which point IEEE Std 830-1993 will be available from the IEEE, 1-800-678-4333.

⁴ISO 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.

Table 1—The relationship of software engineering standards to this recommended practice

Step	Standard	Role
All	IEEE Std 610.12-1990	Terminology
1	IEEE Std 730-1989 IEEE Std 1042-1987 IEEE Std 1058-1988	Description of Quality Assurance Process Description of Configuration Management Process Description of Project Management Process
3	IEEE Std 830-1993 ISO/IEC 9126 : 1991	Format and Content of Software Requirements Specification Definition of Quality Characteristics
4	IEEE Std 1209-1992	Description of Software Evaluation Process
5	IEEE Std 730-1989 IEEE Std 829-1983 IEEE Std 1012-1986 IEEE Std 1016-1987 IEEE Std 1028-1988 IEEE Std 1061-1992 IEEE Std 1063-1987 IEEE Std 1074-1991	Description of Quality Assurance Process Content of Software Test Documentation Format and Content of Software V&V Plan Content of Design Description Description of Review and Audit Processes Description of Software Quality Metric Methodology Format and Content of User Documentation Description of Developing Software Life Cycle Processes
7	IEEE Std 1045-1992 IEEE Std 1058-1988	Description of Software Productivity Metrics Description of Project Management Process
9	IEEE Std 1219-1992	Description of Software Maintenance Process

3. Definitions

The definitions listed below establish meaning in the context of this recommended practice. Other definitions can be found in IEEE Std 610.12-1990. Note that for the purpose of this recommended practice, software includes documentation.

3.1 acquirer: A person or organization that acquires or procures a system or software product (which may be part of a system) from a supplier.

3.2 acquisition: The process of obtaining a system or software product.

3.3 commercial-off-the-shelf (COTS): Software defined by a market-driven need, commercially available, and whose fitness for use has been demonstrated by a broad spectrum of commercial users.

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

3.5 developer: A person or organization that performs development activities (including requirements analysis, design, testing through acceptance) during the software life cycle process.

3.6 modified-off-the-shelf (MOTS): Software product that is already developed and available, usable either “as is” or with modification, and provided by the supplier, acquirer, or a third party.

3.7 request for proposal: A document used by the acquirer as a means to announce intention to potential bidders to acquire a specified system or software product (which may be part of a system).

3.8 software acquisition process: The period of time that begins with the decision to acquire a software product and ends when the product is no longer available for use.

3.9 software acquisition process: The software acquisition process typically includes nine steps associated with planning the organizational strategy, implementing an organization’s process, determining the software requirements, identifying potential suppliers, preparing contract requirements, evaluating proposals and selecting the supplier, managing supplier performance, accepting the software, and using the software.

3.10 software: Computer programs, procedures, and associated documentation and data pertaining to the operation of a computer system (see also IEEE Std 610.12-1990).

3.11 software product: The complete set of computer programs, procedures, and associated documentation and data designated for delivery to a user.

3.12 statement of work: A document used by the acquirer as a means to identify, describe, and specify the tasks to be performed under the contract.

3.13 supplier: A person or organization that enters into a contract with the acquirer for the supply of a software product (which may be part of a system) under the terms of the contract.

4. Introducing the software acquisition process

4.1 Software acquisition life cycle

The software acquisition life cycle represents the period of time that begins with the decision to acquire a software product and ends when the product is no longer available for use. It typically includes a planning phase, contracting phase, product implementation phase, product acceptance phase, and follow-on phase. This life cycle provides an overall framework within which most software acquisitions occur.

The phases in the life cycle are broadly defined by a set of milestones that establish the beginning and ending of each phase. Some phases may have a longer duration than others, or may include more activities than other phases. These phases and their key milestones are:

- a) *Planning phase.* This phase begins when the idea or need is established for acquiring software and ends when the Request for Proposal (RFP) is released.
- b) *Contracting Phase.* After the RFP is released, this phase includes activities necessary to ensure that the supplier's products and services can satisfy the acquirer's quality criteria before signing the contract.
- c) *Product implementation phase.* This phase covers the period from contract signing until the software product has been received. A key activity is monitoring the supplier's efforts to ensure that all work and milestones are satisfactorily completed prior to delivery of the software product.
- d) *Product acceptance phase.* This phase includes all activities necessary to evaluate, test, and accept the software product. It begins when the software product is received and ends when the product is accepted.
- e) *Follow-on phase.* After the software product is accepted, this phase includes using the product to meet the acquirer's objectives and evaluating user satisfaction with the software product, its documentation, and support provided from the supplier. This phase continues until all provisions provided in the contract have been completed or until the software product is no longer available for use.

Each of these phases and their key milestones are summarized in table 2. A special feature of this table includes a listing of the software acquisition process steps associated with each life cycle phase.

Table 2—Software acquisition phase milestones

Phase	Phase initiation milestone	Phase completion milestone	Steps in software acquisition process*
Planning	Develop the idea	Release the RFP	1) Planning organizational strategy 2) Implementing organization's process 3) Determining the software requirements
Contracting	RFP is released	Sign the contract	4) Identifying potential suppliers 5) Preparing contract requirements 6) Evaluating proposals and selecting the supplier
Product implementation	Contract is signed	Receive the software product	7) Managing supplier performance
Product acceptance	Software product is received	Accept the product	8) Accepting the software
Follow-on	Software product is accepted	Product is no longer in use	9) Using the software

*The step numbers referred to in this table correlate to the nine steps outlined in 4.2.

4.2 Nine steps in acquiring quality software

The software acquisition process provides a structure of major acquisition steps that are applicable to the acquisition of either fully developed software or modified-off-the-shelf software. The activities contained in each step all bear upon the development of a software product with the potential for high quality. Other related project objectives may also be improved upon or enhanced through the application of quality principles. Such objectives include on-time delivery and cost-effectiveness. Utilization of the activities in this process is expected to result in the delivery of high-quality, well-documented products.

A typical software acquisition process is provided in figure 1 for understanding of the steps that an acquirer and supplier go through. Checklists have been provided in annex A to assist in establishing a software acquisition process.

While this process may appear to define the acquisition of fully developed software, it can be tailored to fit the acquisition of modified-off-the-shelf software by selecting those activities relevant to that purpose.

The steps in this process include issues that organizations should consider when acquiring fully developed software. Some of the stated principles apply to contracting for any type of service and are covered in general acquisition guidance. However, these principles are included here for completeness and to emphasize that they can indeed be applied to fully developed software contracts as well as to others.

The software acquisition process is divided into nine steps. The steps are:

- a) *Step 1: Planning organizational strategy.* Review acquirer's objectives and develop a strategy for acquiring software.
- b) *Step 2: Implementing organization's process.* Establish a software acquisition process that fits organization's needs for obtaining a quality software product. Include appropriate contracting practices.
- c) *Step 3: Determining the software requirements.* Define the software being acquired and prepare quality and maintenance plans for accepting software supplied by the supplier.
- d) *Step 4: Identifying potential suppliers.* Select potential candidates who will provide documentation for their software, demonstrate their software, and provide formal proposals. Failure to perform any of these actions is basis to reject a potential supplier. Review supplier performance data from previous contracts.
- e) *Step 5: Preparing contract requirements.* Describe the quality of the work to be done in terms of acceptable performance and acceptance criteria, and prepare contract provisions that tie payments to deliverables. Review contract with legal counsel.
- f) *Step 6: Evaluating proposals and selecting the supplier.* Evaluate supplier proposals, select a qualified supplier, and negotiate the contract. Negotiate with an alternate supplier, if necessary.
- g) *Step 7: Managing supplier performance.* Monitor supplier's progress to ensure all milestones are met and to approve work segments. Provide all acquirer deliverables to the supplier when required.
- h) *Step 8: Accepting the software.* Do adequate testing and establish a process for certifying that all discrepancies have been corrected and that all acceptance criteria have been satisfied.
- i) *Step 9: Using the software.* Conduct a follow-up analysis of the software acquisition contract to evaluate contracting practices, record lessons learned, and evaluate user satisfaction with the product. Retain supplier performance data.

These steps may overlap or occur in a different sequence, depending upon the organizational needs. One sequence is shown in figure 1.

Each step in the software acquisition process has certain key inputs and outputs. These are identified in table 3. Note that outputs from various steps are also inputs to subsequent steps. Overall, defining the inputs and outputs provides a better understanding of each of the steps, which are explained in 5.1 through 5.9.

The subsequent clauses provide a solid basis for developing organizational procedures; they might also be helpful for developing ideas in an evaluation group.