IEEE Std 1028-1997

IEEE Standard for Software Reviews

IEEE Computer Society

Sponsored by the
Software Engineering Standards Committee

4 March 1998 SH94592

IEEE Standard for Software Reviews

Sponsor

Software Engineering Standards Committee
of the
IEEE Computer Society

Approved 9 December 1997

IEEE Standards Board

Abstract: This standard defines five types of software reviews, together with procedures required for the execution of each review type. This standard is concerned only with the reviews; it does not define procedures for determining the necessity of a review, nor does it specify the disposition of the results of the review. Review types include management reviews, technical reviews, inspections, walk-throughs, and audits.

Keywords: audit, inspection, review, walk-through

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

ISBN 1-55937-987-1

IEEE Standards documents are developed within 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 societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

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

Secretary, IEEE 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.

Introduction

(This introduction is not part of IEEE Std 1028-1997, IEEE Standard for Software Reviews.)

This Introduction provides the user with the rationale and background of the reviews outlined in this standard and their relationships to other IEEE standards.

Purpose

This standard defines five types of software reviews, together with procedures required for the execution of each review type. This standard is concerned only with the reviews; it does not define procedures for determining the necessity of a review, nor does it specify the disposition of the results of the review. Review types include management reviews, technical reviews, inspections, walk-throughs, and audits.

This standard is meant to be used either in conjunction with other IEEE software engineering standards or as a stand-alone definition of software review procedures. In the latter case, local management must determine the events that precede and follow the actual software reviews.

The need for reviews is described in several other IEEE standards, as well as standards prepared by other standards-writing organizations. IEEE Std 1028-1997 is meant to support these other standards. In particular, reviews required by the following standards can be executed using the procedures described herein:

— IEEE Std 730-1989 [B1][a]
— IEEE Std 828-1990 [B2]
— IEEE Std 1012-1986 [B5]
— IEEE Std 1058.1-1987 [B8]
— IEEE Std 1074-1995 [B10]
— IEEE Std 1219-1992 [B11]
— IEEE Std 1220-1994 [B12]
— IEEE Std 1228-1994 B13]
— IEEE Std 1298-1992 (AS 3563.1-1991) [B14]
— ISO/IEC 12207:1995 [B15]

The use of IEEE Std 1044-1993 [B7] is encouraged as part of the reporting procedures for this standard.

General application intent

This standard applies throughout the scope of any selected software life-cycle model and provides a standard against which software review plans can be prepared and assessed. Maximum benefit can be derived from this standard by planning for its application early in the project life cycle.

This standard for software reviews was written in consideration of both the software and its system operating environment. It can be used where software is the total system entity or where it is part of a larger system. Care should be taken to integrate software review activities into any total system life-cycle planning; software reviews should exist in concert with hardware and computer system reviews to the benefit of the entire system.

Reviews carried out in conformance with this standard may include both personnel internal to the project and customers or acquirers of the product, according to local procedures. Subcontractors may also be included if appropriate.

[a]The numbers in brackets correspond to those of the bibliography in Annex C.

The information obtained during software reviews (particularly inspections) may be of benefit for improving the user's software acquisition, supply, development, operation, and maintenance processes. The use of review data for process improvement is not required by this standard, but their use is strongly encouraged.

Conformance

Conformance to this standard for a specific review type can be claimed when all mandatory actions (indicated by "shall") are carried out as defined in this standard for the review type used. Claims for conformance should be phrased to indicate the review types used; for example, "conforming to IEEE Std 1028-1997 for inspections."

Development procedure

This standard was developed by the Software Engineering Review Working Group. The entire standards writing procedure was carried out via electronic mail.

Participants

At the time this standard was completed, the Software Engineering Review Working Group had the following membership:

J. Dennis Lawrence, *Chair*
Patricia A. Trellue, *Technical Editor*

Frank Ackerman
Leo Beltracchi
Ron Berlack
Antonio Bertolino
Richard J. Blauw
Audrey Brewer
James E. Cardow
Hu Cheng
Pat Daggett
Ronald Dean†
Janet Deeney*†
Claude G. Diderich
Leo G. Egan
Martin Elliot
Jon Fairclough*
Karol Fruehauf
Andrew Gabb
Tom Gilb
Jon Hagar
John Harauz
Hans-Ludwig Hausen
Michael Haux
Herb Hecht
Chuck Howell
Laura Ippolito
Rikkila Juha
George X. Kambic
Myron S. Karasik
Stanley H. Levinson
Michael S. Lines
Jordan Matejceck
Archibald McKinlay
Warren L. Persons†
Peter T. Poon
Christian Reiser
Helmut Sandmayr
Hans Schaefer*
Katsu Shintani
Mel E. Smyre
Julia Stesney
Gina To†
André Villas-Boas
Dolores Wallace
David A. Wheeler
Ron Yun
Tony Zawilski

* Principal writers
† Ballot resolution

The following persons were on the balloting committee:

Leo Beltracchi
Mordechai Ben-Menachem
H. Ronald Berlack
Audrey C. Brewer
Alan L. Bridges
Kathleen L. Briggs
David W. Burnett
Edward R. Byrne
Thomas G. Callaghan
Stuart Ross Campbell
James E. Cardow
Jaya R. Carl
Leslie Chambers
Keith Chan
John P. Chihorek
S. V. Chiyyarath
Antonio M. Cicu
Theo Clarke
Sylvain Clermont
Rosemary Coleman
Darrell Cooksey
Geoff Cozens
Thomas Crowley
Gregory T. Daich
Hillary Davidson
Bostjan K. Derganc
Sanjay Dewal
Michael P. Dewalt
Charles Droz
Robert G. Ebenau
Chrisof Ebert
William Eventoff
Jonathan H. Fairclough
John W. Fendrich
Jay Forster
Kirby Fortenberry
Barry L. Garner
Adel N. Ghannam
Hiranmay Ghosh
Marilyn Ginsberg-Finner
M. Joel Gittleman
John Garth Glynn
Julio Gonzalez-Sanz
Lewis Gray
Lawrence M. Gunther
Jon Hagar
John Harauz
Rob Harker
Herbert Hecht
William Hefley
Manfred Hein
Mark Henley
Umesh P. Hiriyannaiah
John W. Horch
Fabrizio Imelio
George Jackelen
Frank V. Jorgensen
Vladan V. Jovanovic
William S. Junk
George X. Kambic
David W. Kane
Myron S. Karasik
Ron S. Kenett
Judy Kerner
Robert J. Kierzyk
Motti Y. Klein
Dwayne L. Knirk
Shaye Koenig
Joan Kundig
Thomas M. Kurihara
J. Dennis Lawrence
Randal Leavitt
Stanley H. Levinson
Michael Lines
William M. Lively
Dieter Look
David Maibor
Philip P. Mak
Tomoo Matsubara
Scott D. Matthews
Patrick McCray
Sue McGrath
Bret Michael
Alan Miller
Millard Allen Mobley
James W. Moore
Mike Ottewill
Mark Paulk
David E. Peercy
Warren L. Persons
John G. Phippen
Peter T. Poon
Margaretha W. Price
Lawrence S. Przybylski
Kenneth R. Ptack
Terence P. Rout
Andrew P. Sage
Helmut Sandmayr
Stephen R. Schach
Hans Schaefer
David J. Schultz
Gregory D. Schumacher
Robert W. Shillato
Katsutoshi Shintani
Carl A. Singer
James M. Sivak
Alfred R. Sorkowitz
Donald W. Sova
Fred J. Strauss
Michael Surratt
Douglas H. Thiele
Booker Thomas
Carmen J. Trammell
Patricia A. Trellue
Richard D. Tucker
Margaret C. Updike
Theodore J. Urbanowicz
Glenn D. Venables
Dolores Wallace
David A. Wheeler
Camille S. White-Partain
Charles D. Wilson
Paul R. Work
Weider D. Yu
Peter F. Zoll

When the IEEE Standards Board approved this standard on 9 December 1997, it had the following membership:

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

Clyde R. Camp
Stephen L. Diamond
Harold E. Epstein
Donald C. Fleckenstein
Jay Forster*
Thomas F. Garrity
Donald N. Heirman
Jim Isaak
Ben C. Johnson
Lowell Johnson
Robert Kennelly
E. G. "Al" Kiener
Joseph L. Koepfinger*
Stephen R. Lambert
Lawrence V. McCall
L. Bruce McClung
Marco W. Migliaro
Louis-François Pau
Gerald H. Peterson
John W. Pope
Jose R. Ramos
Ronald H. Reimer
Ingo Rüsch
John S. Ryan
Chee Kiow Tan
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal
Alan H. Cookson

Paula M. Kelty
IEEE Standards Project Editor

Contents

IEEE Standard for Software Reviews

1. Overview

1.1 Purpose

The purpose of this standard is to define systematic reviews applicable to software acquisition, supply, development, operation, and maintenance. This standard describes how to carry out a review. Other standards or local management define the context within which a review is performed, and the use made of the results of the review. Software reviews can be used in support of the objectives of project management, system engineering (for example, functional allocation between hardware and software), verification and validation, configuration management, and quality assurance. Different types of reviews reflect differences in the goals of each review type. Systematic reviews are described by their defined procedures, scope, and objectives.

1.2 Scope

This standard provides minimum acceptable requirements for systematic software reviews, where “systematic” includes the following attributes:

a) Team participation
b) Documented results of the review
c) Documented procedures for conducting the review

Reviews that do not meet the requirements of this standard are considered to be nonsystematic reviews. This standard is not intended to discourage or prohibit the use of nonsystematic reviews.

The definitions, requirements, and procedures for the following five types of reviews are included within this standard:

a) Management reviews
b) Technical reviews
c) Inspections
d) Walk-throughs
e) Audits

This standard does not establish the need to conduct specific reviews; that need is defined by other software engineering standards or by local procedures. This standard provides definitions, requirements, and procedures that are applicable to the reviews of software development products throughout the software life cycle.

Users of this standard shall specify where and when this standard applies and any intended deviations from this standard.

It is intended that this standard be used with other software engineering standards that determine the products to be reviewed, the timing of reviews, and the necessity for reviews. This standard is closely aligned with IEEE Std 1012-1986 [B5],[1] but can also be used with IEEE Std 1074-1995 [B10], IEEE Std 730-1989 [B1], ISO/IEC 12207:1995 [B15], and other standards. Use with other standards is described in Annex A. A useful model is to consider IEEE Std 1028-1997 as a subroutine to the other standards. Thus, if IEEE Std 1012-1986 were used to carry out the verification and validation process, the procedure in IEEE Std 1012-1986 could be followed until such time as instructions to carry out a specific review are encountered. At that point, IEEE Std 1028-1997 would be "called" to carry out the review, using the specific review type described herein. Once the review has been completed, IEEE Std 1012-1986 would be returned to for disposition of the results of the review and any additional action required by IEEE Std 1012-1986.

In this model, requirements and quality attributes for the software product are "parameter inputs" to the review and are imposed by the "caller." When the review is finished, the review outputs are "returned" to the "caller" for action. Review outputs typically include anomaly lists and action item lists; the resolution of the anomalies and action items are the responsibility of the "caller."

1.3 Conformance

Conformance to this standard for a specific review type can be claimed when all mandatory actions (indicated by "shall") are carried out as defined in this standard for the review type used. Claims for conformance should be phrased to indicate the review types used; for example, "conforming to IEEE Std 1028-1997 for inspections." The word "shall" is used to express a requirement, "should," to express a recommendation, and "may," to express alternative or optional methods of satisfying a requirement.

1.4 Organization of standard

Clauses 4–8 of this standard provide guidance and descriptions for the five types of systematic reviews addressed by this standard. Each of these clauses contains the following information:

a) *Introduction.* Describes the objectives of the systematic review and provides an overview of the systematic review procedures.
b) *Responsibilities.* Defines the roles and responsibilities needed for the systematic review.
c) *Input.* Describes the requirements for input needed by the systematic review.
d) *Entry criteria.* Describes the criteria to be met before the systematic review can begin, including
    1) Authorization
    2) Initiating event
e) *Procedures.* Details the procedures for the systematic review, including
    1) Planning the review
    2) Overview of procedures
    3) Preparation
    4) Examination/evaluation/recording of results
    5) Rework/follow-up
f) *Exit criteria.* Describes the criteria to be met before the systematic review can be considered complete.
g) *Output.* Describes the minimum set of deliverables to be produced by the systematic review.

[1]The numbers in brackets correspond to those of the bibliography in Annex C.

1.5 Application of standard

The procedures and terminology defined in this standard apply to software acquisition, supply, development, operation, and maintenance processes requiring systematic reviews. Systematic reviews are performed on a software product as required by other standards or local procedures.

The term “software product” is used in this standard in a very broad sense. Examples of software products include, but are not limited to, the following:

a) Anomaly reports
b) Audit reports
c) Back up and recovery plans
d) Build procedures
e) Contingency plans
f) Contracts
g) Customer or user representative complaints
h) Disaster plans
i) Hardware performance plans
j) Inspection reports
k) Installation plans
l) Installation procedures
m) Maintenance manuals
n) Maintenance plans
o) Management review reports
p) Operations and user manuals
q) Procurement and contracting methods
r) Progress reports
s) Release notes
t) Reports and data (for example, review, audit, project status, anomaly reports, test data)
u) Request for proposal
v) Risk management plans
w) Software configuration management plans (see IEEE Std 828-1990 [B2])
x) Software design descriptions (see IEEE Std 1016-1987 [B6])
y) Software project management plans (see IEEE Std 1058-1987 [B8])
z) Software quality assurance plans (see IEEE Std 730-1989 [B1])
aa) Software requirements specifications (see IEEE Std 830-1993 [B4])
ab) Software safety plans (see IEEE 1228-1994 [B13])
ac) Software test documentation (see IEEE Std 829-1983 [B3])
ad) Software user documentation (see IEEE Std 1063-1987 [B9])
ae) Software verification and validation plans (see IEEE Std 1012-1986 [B5])
af) Source code
ag) Standards, regulations, guidelines, and procedures
ah) System build procedures
ai) Technical review reports
aj) Vendor documents
ak) Walk-through reports

This standard permits reviews that are held by means other than physically meeting in a single location. Examples include telephone conferences, video conferences, and other means of group electronic communication. In such cases the communication means should be defined in addition to the meeting places, and all other review requirements remain applicable.

In order to make use of this standard to carry out a software review, first decide the objective of the review. Next, select an appropriate review type using the guidance in Annex B or a local procedure. Then follow the procedure described in the appropriate clause (4–8) of this standard.

2. References

This standard shall be used in conjunction with the following publications. If the following publications are superseded by an approved revision, the revision shall apply. (Additional standards that may be used to prepare software products that are the subject of reviews are cited in a bibliography in Annex C.)

IEEE Std 100-1996, The IEEE Standard Dictionary of Electrical and Electronics Terms, Sixth Edition.

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

3. Definitions

For purposes of this standard, the following terms and definitions apply. IEEE Std 610.12-1990[2] and IEEE Std 100-1996 should be consulted for terms not defined in this clause.

Six of the terms given here are defined in other IEEE software engineering standards. The definition of the term “anomaly” is identical to that given in IEEE Std 1044-1993 [B7]. The terms “audit,” “inspection,” “review,” “software product,” and “walk-through” are all defined in IEEE Std 610.12-1990; however, some minor modifications have been made to those definitions to more closely match the content of this standard, as explained in the succeeding paragraph.

IEEE Std 610.12-1990 uses different terms for the object of a review: audits and reviews are defined therein in terms of “work products,” inspections are defined in terms of “development products,” and walk-throughs are defined in terms of “segment of documentation or code.” “Work products” are not defined in IEEE Std 610.12-1990. Since “software product” is defined therein, and it is desirable to use a single term in this standard, a change in terminology was made. Since software products being reviewed are not limited to those “designated for delivery to a user,” that phrase was dropped from the definition of “software product.” The definition of “inspection” has been changed considerably. No other changes to the definitions from IEEE Std 610.12-1990 were made.

3.1 anomaly: Any condition that deviates from expectations based on requirements specifications, design documents, user documents, standards, etc., or from someone’s perceptions or experiences. Anomalies may be found during, but not limited to, the review, test, analysis, compilation, or use of software products or applicable documentation.

3.2 audit: An independent examination of a software product, software process, or set of software processes to assess compliance with specifications, standards, contractual agreements, or other criteria.

3.3 inspection: A visual examination of a software product to detect and identify software anomalies, including errors and deviations from standards and specifications. Inspections are peer examinations led by impartial facilitators who are trained in inspection techniques. Determination of remedial or investigative action for an anomaly is a mandatory element of a software inspection, although the solution should not be determined in the inspection meeting.

[2]Information on references can be found in Clause 2.

3.4 management review: A systematic evaluation of a software acquisition, supply, development, operation, or maintenance process performed by or on behalf of management that monitors progress, determines the status of plans and schedules, confirms requirements and their system allocation, or evaluates the effectiveness of management approaches used to achieve fitness for purpose.

3.5 review: A process or meeting during which a software product is presented to project personnel, managers, users, customers, user representatives, or other interested parties for comment or approval.

3.6 software product: (A) A complete set of computer programs, procedures, and associated documentation and data. (B) One or more of the individual items in (A).

3.7 technical review: A systematic evaluation of a software product by a team of qualified personnel that examines the suitability of the software product for its intended use and identifies discrepancies from specifications and standards. Technical reviews may also provide recommendations of alternatives and examination of various alternatives.

3.8 walk-through: A static analysis technique in which a designer or programmer leads members of the development team and other interested parties through a software product, and the participants ask questions and make comments about possible errors, violation of development standards, and other problems.

4. Management reviews

4.1 Introduction

The purpose of a management review is to monitor progress, determine the status of plans and schedules, confirm requirements and their system allocation, or evaluate the effectiveness of management approaches used to achieve fitness for purpose. Management reviews support decisions about corrective actions, changes in the allocation of resources, or changes to the scope of the project.

Management reviews are carried out by, or on behalf of, the management personnel having direct responsibility for the system. Management reviews identify consistency with and deviations from plans, or adequacies and inadequacies of management procedures. This examination may require more than one meeting. The examination need not address all aspects of the product.

Examples of software products subject to management review include, but are not limited to

a) Anomaly reports
b) Audit reports
c) Back-up and recovery plans
d) Contingency plans
e) Customer or user representative complaints
f) Disaster plans
g) Hardware performance plans
h) Installation plans
i) Maintenance plans
j) Procurement and contracting methods
k) Progress reports
l) Risk management plans
m) Software configuration management plans
n) Software project management plans
o) Software quality assurance plans
p) Software safety plans

q) Software verification and validation plans
r) Technical review reports
s) Software product analyses
t) Verification and validation reports

4.2 Responsibilities

Management reviews are carried out by, or on behalf of, the management personnel having direct responsibility for the system. Technical knowledge may be necessary to conduct a successful management review. Management reviews shall be performed by the available personnel who are best qualified to evaluate the software product.

The following roles shall be established for the management review:

a) Decision maker
b) Review leader
c) Recorder
d) Management staff
e) Technical staff

The following roles may also be established for the management review:

f) Other team members
g) Customer or user representative
h) Individual participants may act in more than one role

4.2.1 Decision maker

The decision maker is the person for whom the management review is conducted. The decision maker shall determine if the review objectives have been met.

4.2.2 Review leader

The review leader shall be responsible for administrative tasks pertaining to the review, shall be responsible for planning and preparation as described in 4.5.2 and 4.5.4, shall ensure that the review is conducted in an orderly manner and meets its objectives, and shall issue the review outputs as described in 4.7.

4.2.3 Recorder

The recorder shall document anomalies, action items, decisions, and recommendations made by the review team.

4.2.4 Management staff

Management staff assigned to carry out management reviews are responsible for active participation in the review. Managers responsible for the system as a whole have additional responsibilities as defined in 4.5.1.

4.2.5 Technical staff

The technical staff shall provide the information necessary for the management staff to fulfill its responsibilities.

4.2.6 Customer or user representative

The role of the customer or user representative should be determined by the review leader prior to the review.

4.3 Input

Input to the management review shall include the following:

a) A statement of objectives for the management review
b) The software product being evaluated
c) Software project management plan
d) Status, relative to plan, of the software product completed or in progress
e) Current anomalies or issues list
f) Documented review procedures

Input to the management review should also include the following:

g) Status of resources, including finance, as appropriate
h) Relevant review reports
i) Any regulations, standards, guidelines, plans, or procedures against which the software product should be evaluated
j) Anomaly categories (See IEEE Std 1044-1993 [B7])

Additional reference material may be made available by the individuals responsible for the software product when requested by the review leader.

4.4 Entry criteria

4.4.1 Authorization

The need for conducting management reviews should initially be established in the appropriate project planning documents, as listed in 4.1. Under these plans, completion of a specific software product or completion of an activity may initiate a management review. In addition to those management reviews required by a specific plan, other management reviews may be announced and held at the request of software quality management, functional management, project management, or the customer or user representative, according to local procedures.

4.4.2 Preconditions

A management review shall be conducted only when both of the following conditions have been met:

a) A statement of objectives for the review is established by the management personnel for whom the review is being carried out
b) The required review inputs are available

4.5 Procedures

4.5.1 Management preparation

Managers shall ensure that the review is performed as required by applicable standards and procedures and by requirements mandated by law, contract, or other policy. To this end, managers shall

a) Plan time and resources required for reviews, including support functions, as required in IEEE Std 1058.1-1987 [B8] or other appropriate standards
b) Provide funding and facilities required to plan, define, execute, and manage the reviews
c) Provide training and orientation on review procedures applicable to a given project
d) Ensure appropriate levels of expertise and knowledge sufficient to comprehend the software product under review
e) Ensure that planned reviews are conducted
f) Act on review team recommendations in a timely manner

4.5.2 Planning the review

The review leader shall be responsible for the following activities:

a) Identify, with appropriate management support, the review team
b) Assign specific responsibilities to the review team members
c) Schedule and announce the meeting
d) Distribute review materials to participants, allowing adequate time for their preparation
e) Set a timetable for distribution of review material, the return of comments, and forwarding of comments to the author for disposition

4.5.3 Overview of review procedures

A qualified person should present an overview session for the review team when requested by the review leader. This overview may occur as part of the review meeting (see 4.5.6) or as a separate meeting.

4.5.4 Preparation

Each review team member shall examine the software product and other review inputs prior to the review meeting. Anomalies detected during this examination should be documented and sent to the review leader. The review leader should classify anomalies to ensure that review meeting time is used most effectively. The review leader should forward the anomalies to the author of the software product for disposition.

4.5.5 Examination

The management review shall consist of one or more meetings of the review team. The meetings shall accomplish the following goals:

a) Review the objectives of the management review
b) Evaluate the software product under review against the review objectives
c) Evaluate project status, including the status of plans and schedules
d) Review anomalies identified by the review team prior to the review
e) Generate a list of action items, emphasizing risks
f) Document the meeting

The meetings should accomplish the following goals as appropriate:

g) Evaluate the risk issues that may jeopardize the success of the project
h) Confirm software requirements and their system allocation
i) Decide the course of action to be taken or recommendations for action
j) Identify other issues that should be addressed

4.5.6 Rework/follow-up

The review leader shall verify that the action items assigned in the meeting are closed.

4.6 Exit criteria

The management review shall be considered complete when the activities listed in 4.5.5 have been accomplished and the output described in 4.7 exists.

4.7 Output

The output from the management review shall be documented evidence that identifies

a) The project being reviewed
b) The review team members
c) Review objectives
d) Software product reviewed
e) Specific inputs to the review
f) Action item status (open, closed), ownership and target date (if open) or completion date (if closed)
g) A list of anomalies identified by the review team that must be addressed for the project to meet its goals

Although this standard sets minimum requirements for the content of the documented evidence, it is left to local procedures to prescribe additional content, format requirements, and media.

5. Technical reviews

5.1 Introduction

The purpose of a technical review is to evaluate a software product by a team of qualified personnel to determine its suitability for its intended use and identify discrepancies from specifications and standards. It provides management with evidence to confirm whether

a) The software product conforms to its specifications
b) The software product adheres to regulations, standards, guidelines, plans, and procedures applicable to the project
c) Changes to the software product are properly implemented and affect only those system areas identified by the change specification

Technical reviews may also provide the recommendation and examination of various alternatives, which may require more than one meeting. The examination need not address all aspects of the product.

Examples of software products subject to technical review include, but are not limited to

a) Software requirements specification
b) Software design description
c) Software test documentation
d) Software user documentation
e) Maintenance manual
f) System build procedures
g) Installation procedures
h) Release notes

5.2 Responsibilities

The following roles shall be established for the technical review:

a) Decision maker
b) Review leader
c) Recorder
d) Technical staff

The following roles may also be established for the technical review:

e) Management staff
f) Other team members
g) Customer or user representative

Individual participants may act in more than one role.

5.2.1 Decision maker

The decision maker is the person for whom the technical review is conducted. The decision maker shall determine if the review objectives have been met.

5.2.2 Review leader

The review leader shall be responsible for the review. This responsibility includes performing administrative tasks pertaining to the review, ensuring that the review is conducted in an orderly manner, and ensuring that the review meets its objectives. The review leader shall issue the review outputs as described in 5.7.

5.2.3 Recorder

The recorder shall document anomalies, action items, decisions, and recommendations made by the review team.

5.2.4 Technical staff

The technical staff shall actively participate in the review and evaluation of the software product.

5.2.5 Management staff

The management staff may participate in the technical review for the purpose of identifying issues that require management resolution.

5.2.6 Customer or user representative

The role of the customer or user representative should be determined by the review leader prior to the review.

5.3 Input

Input to the technical review shall include the following:

a) A statement of objectives for the technical review
b) The software product being examined
c) Software project management plan

d) Current anomalies or issues list for the software product
e) Documented review procedures

Input to the technical review should also include the following:

f) Relevant review reports
g) Any regulations, standards, guidelines, plans, and procedures against which the software product is to be examined
h) Anomaly categories (See IEEE Std 1044-1993 [B7])

Additional reference material may be made available by the individuals responsible for the software product when requested by the review leader.

5.4 Entry criteria

5.4.1 Authorization

The need for conducting technical reviews of a software product shall be defined by project planning documents. In addition to those technical reviews required by a specific plan, other technical reviews may be announced and held at the request of functional management, project management, software quality management, systems engineering, or software engineering according to local procedures. Technical reviews may be required to evaluate impacts of hardware anomalies or deficiencies on the software product.

5.4.2 Preconditions

A technical review shall be conducted only when both of the following conditions have been met:

a) A statement of objectives for the review is established
b) The required review inputs are available

5.5 Procedures

5.5.1 Management preparation

Managers shall ensure that the review is performed as required by applicable standards and procedures and by requirements mandated by law, contract, or other policy. To this end, managers shall

a) Plan time and resources required for reviews, including support functions, as required in IEEE Std 1058.1-1987 [B8] or other appropriate standards
b) Provide funding and facilities required to plan, define, execute, and manage the reviews
c) Provide training and orientation on review procedures applicable to a given project
d) Ensure that review team members possess appropriate levels of expertise and knowledge sufficient to comprehend the software product under review
e) Ensure that planned reviews are conducted
f) Act on review team recommendations in a timely manner

5.5.2 Planning the review

The review leader shall be responsible for the following activities:

a) Identify, with appropriate management support, the review team
b) Assign specific responsibilities to the review team members
c) Schedule and announce the meeting place

d) Distribute review materials to participants, allowing adequate time for their preparation
e) Set a timetable for distribution of review material, the return of comments and forwarding of comments to the author for disposition

As a part of the planning procedure, the review team shall determine if alternatives are to be discussed at the review meeting. Alternatives may be discussed at the review meeting, afterwards in a separate meeting, or left to the author of the software product to resolve.

5.5.3 Overview of review procedures

A qualified person should present an overview of the review procedures for the review team when requested by the review leader. This overview may occur as a part of the review meeting (see 5.5.6) or as a separate meeting.

5.5.4 Overview of the software product

A technically qualified person should present an overview of the software product for the review team when requested by the review leader. This overview may occur either as a part of the review meeting (see 5.5.6) or as a separate meeting.

5.5.5 Preparation

Each review team member shall examine the software product and other review inputs prior to the review meeting. Anomalies detected during this examination should be documented and sent to the review leader. The review leader should classify anomalies to ensure that review meeting time is used most effectively. The review leader should forward the anomalies to the author of the software product for disposition.

The review leader shall verify that team members are prepared for the technical review. The review leader should gather individual preparation times and record the total. The review leader shall reschedule the meeting if the team members are not adequately prepared.

5.5.6 Examination

During the technical review the review team shall hold one or more meetings. The meetings shall accomplish the following goals:

a) Decide on the agenda for evaluating the software product and anomalies
b) Evaluate the software product
c) Determine if
    1) The software product is complete;
    2) The software product conforms to the regulations, standards, guidelines, plans and procedures applicable to the project;
    3) Changes to the software product are properly implemented and affect only the specified areas;
    4) The software product is suitable for its intended use;
    5) The software product is ready for the next activity;
    6) Hardware anomalies or specification discrepancies exist
d) Identify anomalies
e) Generate a list of action items, emphasizing risks
f) Document the meeting

After the software product has been reviewed, documentation shall be generated to document the meeting, list anomalies found in the software product, and describe any recommendations to management.

When anomalies are sufficiently critical or numerous, the review leader should recommend that an additional review be applied to the modified software product. This, at a minimum, should cover product areas changed to resolve anomalies as well as side effects of those changes.

5.5.7 Rework/follow-up

The review leader shall verify that the action items assigned in the meeting are closed.

5.6 Exit criteria

A technical review shall be considered complete when the activities listed in 5.5.6 have been accomplished, and the output described in 5.7 exists.

5.7 Output

The output from the technical review shall consist of documented evidence that identifies

a) The project being reviewed
b) The review team members
c) The software product reviewed
d) Specific inputs to the review
e) Review objectives and whether they were met
f) A list of resolved and unresolved software product anomalies
g) A list of unresolved system or hardware anomalies or specification action items
h) A list of management issues
i) Action item status (open, closed), ownership and target date (if open), or completion date (if closed)
j) Any recommendations made by the review team on how to dispose of unresolved issues and anomalies
k) Whether the software product meets the applicable regulations, standards, guidelines, plans, and procedures without deviations

Although this standard sets minimum requirements for the content of the documented evidence, it is left to local procedures to prescribe additional content, format requirements, and media.

6. Inspections

6.1 Introduction

The purpose of an inspection is to detect and identify software product anomalies. This is a systematic peer examination that

a) Verifies that the software product satisfies its specifications
b) Verifies that the software product satisfies specified quality attributes
c) Verifies that the software product conforms to applicable regulations, standards, guidelines, plans, and procedures
d) Identifies deviations from standards and specifications
e) Collects software engineering data (for example, anomaly and effort data) (optional)
f) Uses the collected software engineering data to improve the inspection process itself and its supporting documentation (for example, checklists) (optional)

Inspections consist of three to six participants. An inspection is led by an impartial facilitator who is trained in inspection techniques. Determination of remedial or investigative action for an anomaly is a mandatory element of a software inspection, although the resolution should not occur in the inspection meeting. Collection of data for the purpose of analysis and improvement of software engineering procedures (including all review procedures) is strongly recommended but is not a mandatory element of software inspections.

Examples of software products subject to inspections include, but are not limited to

a) Software requirements specification
b) Software design description
c) Source code
d) Software test documentation
e) Software user documentation
f) Maintenance manual
g) System build procedures
h) Installation procedures
i) Release notes

6.2 Responsibilities

The following roles shall be established for the inspection:

a) Inspection leader
b) Recorder
c) Reader
d) Author
e) Inspector

All participants in the review are inspectors. The author shall not act as inspection leader and should not act as reader or recorder. Other roles may be shared among the team members. Individual participants may act in more than one role.

Individuals holding management positions over any member of the inspection team shall not participate in the inspection.

6.2.1 Inspection leader

The inspection leader shall be responsible for administrative tasks pertaining to the inspection, shall be responsible for planning and preparation as described in 6.5.2 and 6.5.4, shall ensure that the inspection is conducted in an orderly manner and meets its objectives, should be responsible for collecting inspection data (if appropriate), and shall issue the inspection output as described in 6.7.

6.2.2 Recorder

The recorder shall document anomalies, action items, decisions, and recommendations made by the inspection team. The recorder should record inspection data required for process analysis. The inspection leader may be the recorder.

6.2.3 Reader

The reader shall lead the inspection team through the software product in a comprehensive and logical fashion, interpreting sections of the work (for example, generally paraphrasing groups of 1–3 lines), and highlighting important aspects.

6.2.4 Author

The author shall be responsible for the software product meeting its inspection entry criteria, for contributing to the inspection based on special understanding of the software product, and for performing any rework required to make the software product meet its inspection exit criteria.

6.2.5 Inspector

Inspectors shall identify and describe anomalies in the software product. Inspectors shall be chosen to represent different viewpoints at the meeting (for example, sponsor, requirements, design, code, safety, test, independent test, project management, quality management, and hardware engineering). Only those viewpoints pertinent to the inspection of the product should be present.

Some inspectors should be assigned specific review topics to ensure effective coverage. For example, one inspector may focus on conformance with a specific standard or standards, another on syntax, another for overall coherence. These roles should be assigned by the inspection leader when planning the inspection, as provided in 6.5.2 (b).

6.3 Input

Input to the inspection shall include the following:

a) A statement of objectives for the inspection
b) The software product to be inspected
c) Documented inspection procedure
d) Inspection reporting forms
e) Current anomalies or issues list

Input to the inspection may also include the following:

f) Inspection checklists
g) Any regulations, standards, guidelines, plans, and procedures against which the software product is to be inspected
h) Hardware product specifications
i) Hardware performance data
j) Anomaly categories (see IEEE Std 1044-1993 [B7])

Additional reference material may be made available by the individuals responsible for the software product when requested by the inspection leader.

6.4 Entry criteria

6.4.1 Authorization

Inspections shall be planned and documented in the appropriate project planning documents (for example, the overall project plan, or software verification and validation plan).

Additional inspections may be conducted during acquisition, supply, development, operation, and maintenance of the software product at the request of project management, quality management, or the author, according to local procedures.

6.4.2 Preconditions

An inspection shall be conducted only when both of the following conditions have been met:

a) A statement of objectives for the inspection is established.
b) The required inspection inputs are available.

6.4.3 Minimum entry criteria

An inspection shall not be conducted until all of the following events have occurred, unless there is a documented rationale, accepted by management, for exception from these provisions:

a) The software product that is to be inspected is complete and conforms to project standards for content and format.
b) Any automated error-detecting tools (such as spell-checkers and compilers) required for the inspection are available.
c) Prior milestones are satisfied as identified in the appropriate planning documents.
d) Required supporting documentation is available.
e) For a re-inspection, all items noted on the anomaly list that affect the software product under inspection are resolved.

6.5 Procedures

6.5.1 Management preparation

Managers shall ensure that the inspection is performed as required by applicable standards and procedures and by requirements mandated by law, contract, or other policy. To this end, managers shall

a) Plan time and resources required for inspection, including support functions, as required in IEEE Std 1058.1-1987 [B8] or other appropriate standards
b) Provide funding and facilities required to plan, define, execute, and manage the inspection
c) Provide training and orientation on inspection procedures applicable to a given project
d) Ensure that review team members possess appropriate levels of expertise and knowledge sufficient to comprehend the software product under inspection
e) Ensure that planned inspections are conducted
f) Act on inspection team recommendations in a timely manner

6.5.2 Planning the inspection

The author shall assemble the inspection materials for the inspection leader.

The inspection leader shall be responsible for the following activities:

a) Identifying, with appropriate management support, the inspection team
b) Assigning specific responsibilities to the inspection team members
c) Scheduling the meeting and selecting the meeting place
d) Distributing inspection materials to participants, and allowing adequate time for their preparation
e) Setting a timetable for distribution of inspection material and for the return of comments and forwarding of comments to the author for disposition

As a part of the planning procedure, the inspection team shall determine if alternatives are to be discussed at the inspection meeting. Alternatives may be discussed at the inspection meeting, afterwards in a separate meeting, or left to the authors of the software product to resolve.

6.5.3 Overview of inspection procedures

The author should present an overview of the software product to be inspected. This overview should be used to introduce the inspectors to the software product. The overview may be attended by other project personnel who could profit from the presentation.

Roles shall be assigned by the inspection leader. The inspection leader shall answer questions about any checklists and the role assignments and should present inspection data such as minimal preparation times and the typical number of anomalies found in past similar products.

6.5.4 Preparation

Each inspection team member shall examine the software product and other review inputs prior to the review meeting. Anomalies detected during this examination shall be documented and sent to the inspection leader. The inspection leader should classify anomalies to ensure that inspection meeting time is used effectively. The inspection leader should forward the anomalies to the author of the software product for disposition.

The inspection leader or reader shall specify a suitable order in which the software product will be inspected (such as sequential, hierarchical, data flow, control flow, bottom up, or top down). The reader shall ensure that he or she is able to present the software product at the inspection meeting.

6.5.5 Examination

The inspection meeting shall follow this agenda:

6.5.5.1 Introduce meeting

The inspection leader shall introduce the participants and describe their roles. The inspection leader shall state the purpose of the inspection and should remind the inspectors to focus their efforts toward anomaly detection, not resolution. The inspection leader should remind the inspectors to direct their remarks to the reader and to comment only on the software product, not their author. Inspectors may pose questions to the author regarding the software product. The inspection leader shall resolve any special procedural questions raised by the inspectors.

6.5.5.2 Establish preparedness

The inspection leader shall verify that inspectors are prepared for the inspection. The inspection leader shall reschedule the meeting if the inspectors are not adequately prepared. The inspection leader should gather individual preparation times and record the total in the inspection documentation.

6.5.5.3 Review general items

Anomalies referring to the software product in general (and thus not attributable to a specific instance or location) shall be presented to the inspectors and recorded.

6.5.5.4 Review software product and record anomalies

The reader shall present the software product to the inspection team. The inspection team shall examine the software product objectively and thoroughly, and the inspection leader shall focus this part of the meeting on creating the anomaly list. The recorder shall enter each anomaly, location, description, and classification on the anomaly list. IEEE Std 1044-1993 [B7] may be used to classify anomalies. During this time, the author shall answer specific questions and contribute to anomaly detection based on the author's special understanding of the software product. If there is disagreement about an anomaly, the potential anomaly shall be logged and marked for resolution at the end of the meeting.

6.5.5.5 Review the anomaly list

At the end of the inspection meeting, the inspection leader should have the anomaly list reviewed with the team to ensure its completeness and accuracy. The inspection leader should allow time to discuss every anomaly where disagreement occurred. The inspection leader should not allow the discussion to focus on resolving the anomaly but on clarifying what constitutes the anomaly.

6.5.5.6 Make exit decision

The purpose of the exit decision is to bring an unambiguous closure to the inspection meeting. The exit decision shall determine if the software product meets the inspection exit criteria and shall prescribe any appropriate rework and verification. Specifically, the inspection team shall identify the software product disposition as one of the following:

a) *Accept with no or minor rework.* The software product is accepted as is or with only minor rework (for example, that would require no further verification).
b) *Accept with rework verification.* The software product is to be accepted after the inspection leader or a designated member of the inspection team (other than the author) verifies rework.
c) *Re-inspect.* Schedule a re-inspection to verify rework. At a minimum, a re-inspection shall examine the software product areas changed to resolve anomalies identified in the last inspection, as well as side effects of those changes.

6.5.6 Rework/follow-up

The inspection leader shall verify that the action items assigned in the meeting are closed.

6.6 Exit criteria

An inspection shall be considered complete when the activities listed in 6.5.5 have been accomplished, and the output described in 6.7 exists.

6.7 Output

The output of the inspection shall be documented evidence that identifies

a) The project being inspected
b) The inspection team members
c) The inspection meeting duration
d) The software product inspected
e) The size of the materials inspected (for example, the number of text pages)
f) Specific inputs to the inspection
g) Inspection objectives and whether they were met
h) The anomaly list, containing each anomaly location, description, and classification
i) The inspection anomaly summary listing the number of anomalies identified by each anomaly category
j) The disposition of the software product
k) An estimate of the rework effort and rework completion date

The output of the inspection should include the following documentation:

l) The total preparation time of the inspection team

Although this standard sets minimum requirements for the content of the documented evidence, it is left to local procedures to prescribe additional content, format requirements, and media.

6.8 Data collection recommendations

Inspections should provide data for the analysis of the quality of the software product, the effectiveness of the acquisition, supply, development, operation and maintenance processes, and the efficiency of the inspection itself. In order to maintain the effectiveness of inspections, data should not be used to evaluate the performance of individuals. To enable these analyses, anomalies that are identified at an inspection meeting should be classified in accordance with 6.8.1 through 6.8.3.

Inspection data should contain the identification of the software product, the date and time of the inspection, the inspection leader, the preparation and inspection times, the volume of the materials inspected, and the disposition of the inspected software product. The capture of this information can be used to optimize local guidance for inspections.

The management of inspection data requires a capability to store, enter, access, update, summarize, and report categorized anomalies. The frequency and types of the inspection analysis reports, and their distribution, are left to local standards and procedures.

6.8.1 Anomaly classification

Anomalies may be classified by technical type according to, for example, IEEE Std 1044-1993 [B7].

6.8.2 Anomaly classes

Anomaly classes provide evidence of nonconformance and may be categorized, for example, as

a) Missing
b) Extra (superfluous)
c) Ambiguous
d) Inconsistent
e) Improvement desirable
f) Not conforming to standards
g) Risk-prone, i.e., the review finds that, although an item was not shown to be “wrong,” the approach taken involves risks (and there are known safer alternative methods)
h) Factually incorrect
i) Not implementable (e.g., because of system constraints or time constraints)
j) Editorial

6.8.3 Anomaly ranking

Anomalies may be ranked by potential impact on the software product, for example, as

a) *Major.* Anomalies that would result in failure of the software product or an observable departure from specification.
b) *Minor.* Anomalies that deviate from relevant specifications but will not cause failure of the software product or an observable departure in performance.

6.9 Improvement

Inspection data should be analyzed regularly in order to improve the inspection itself, and the software activities used to produce software products. Frequently occurring anomalies may be included in the inspection

checklists or role assignments. The checklists themselves should also be inspected regularly for superfluous or misleading questions. The preparation times, meeting times, and number of participants should be analyzed to determine connections between preparation rate, meeting rate, and number and severity of anomalies found.

A “chief inspector” role should exist. The chief inspector acts as the inspection owner, and collects and feeds back data about the inspection. This chief inspector should be responsible for the proposed follow-up on the inspection itself.

7. Walk-throughs

7.1 Introduction

The purpose of a systematic walk-through is to evaluate a software product. A walk-through may be held for the purpose of educating an audience regarding a software product. The major objectives are to

a) Find anomalies
b) Improve the software product
c) Consider alternative implementations
d) Evaluate conformance to standards and specifications

Other important objectives of the walk-through include exchange of techniques and style variations and training of the participants. A walk-through may point out several deficiencies (for example, efficiency and readability problems in the software product, modularity problems in design or code, or untestable specifications).

Examples of software products subject to walk-throughs include, but are not limited to,

a) Software requirements specification
b) Software design description
c) Source code
d) Software test documentation
e) Software user documentation
f) Maintenance manual
g) System build procedures
h) Installation procedures
i) Release notes

7.2 Responsibilities

The following roles shall be established for the walk-through:

a) Walk-through leader
b) Recorder
c) Author
d) Team member

For a review to be considered a systematic walk-through, a team of at least two members shall be assembled. Roles may be shared among the team members. The walk-through leader or the author may serve as the recorder. The walk-through leader may be the author.

Individuals holding management positions over any member of the walk-through team shall not participate in the walk-through.

7.2.1 Walk-through leader

The walk-through leader shall conduct the walk-through, shall handle the administrative tasks pertaining to the walk-through (such as distributing documents and arranging the meeting), and shall ensure that the walk-through is conducted in an orderly manner. The walk-through leader shall prepare the statement of objectives to guide the team through the walk-through. The walk-through leader shall ensure that the team arrives at a decision or identified action for each discussion item, and shall issue the walk-through output as described in 7.7.

7.2.2 Recorder

The recorder shall note all decisions and identified actions arising during the walk-through meeting. In addition, the recorder should note all comments made during the walk-through that pertain to anomalies found, questions of style, omissions, contradictions, suggestions for improvement, or alternative approaches.

7.2.3 Author

The author should present the software product in the walk-through.

7.3 Input

Input to the walk-through shall include the following:

a) A statement of objectives for the walk-through
b) The software product being examined
c) Standards that are in effect for the acquisition, supply, development, operation, and/or maintenance of the software product

Input to the walk-through may also include the following:

d) Any regulations, standards, guidelines, plans, and procedures against which the software product is to be inspected
e) Anomaly categories (see IEEE Std 1044-1993 [B7])

7.4 Entry criteria

7.4.1 Authorization

The need for conducting walk-throughs shall be established in the appropriate project planning documents. Additional walk-throughs may be conducted during acquisition, supply, development, operation, and maintenance of the software product at the request of project management, quality management, or the author, according to local procedures.

7.4.2 Preconditions

A walk-through shall be conducted only when both of the following conditions have been met:

a) A statement of objectives for the review is established by the management personnel for whom the review is being carried out.
b) The required review inputs are available.

7.5 Procedures

7.5.1 Management preparation

Managers shall ensure that the walk-through is performed as required by applicable standards and procedures and by requirements mandated by law, contract, or other policy. To this end, managers shall

a) Plan time and resources required for walk-throughs, including support functions, as required in IEEE Std 1058.1-1987 [B8] or other appropriate standards
b) Provide funding and facilities required to plan, define, execute, and manage the walk-through
c) Provide training and orientation on walk-through procedures applicable to a given project
d) Ensure that walk-through team members possess appropriate levels of expertise and knowledge sufficient to comprehend the software product
e) Ensure that planned walk-throughs are conducted
f) Act on walk-through team recommendations in a timely manner

7.5.2 Planning the walk-through

The walk-through leader shall be responsible for the following activities:

a) Identifying the walk-through team
b) Scheduling the meeting and selecting the meeting place
c) Distributing necessary input materials to participants, and allowing adequate time for their preparation

7.5.3 Overview

An overview presentation should be made by the author as part of the walk-through meeting.

7.5.4 Preparation

The walk-through leader shall distribute the software product and convene a walk-through meeting. Team members shall prepare for the meeting by examining the software product and preparing a list of items for discussion in the meeting. These items should be divided into two categories: general and specific. General items apply to the whole product; specific items apply to a part of it.

Each walk-through team member shall examine the software product and other review inputs prior to the review meeting. Anomalies detected during this examination shall be documented and sent to the walk-through leader. The walk-through leader should classify anomalies to ensure that walk-through meeting time is used effectively. The walk-through leader should forward the anomalies to the author of the software product for disposition.

The author or walk-through leader shall specify a suitable order in which the software product will be inspected (such as sequential, hierarchical, data flow, control flow, bottom up, or top down).

7.5.5 Examination

The walk-through leader shall introduce the participants and describe their roles. The walk-through leader shall state the purpose of the walk-through and should remind the team members to focus their efforts toward anomaly detection, not resolution. The walk-through leader should remind the team members to comment only on the software product, not its author. Team members may pose questions to the author regarding the software product. The walk-through leader shall resolve any special procedural questions raised by the team members.

The author shall present an overview of the software product under review. This is followed by a general discussion during which team members raise their general items. After the general discussion, the author serially presents the software product in detail (hence the name “walk-through”). Team members raise their specific items when the author reaches them in the presentation. New items may be raised during the meeting. The walk-through leader coordinates discussion and guides the meeting to a decision or identified action on each item. The recorder notes all recommendations and required actions.

During the walk-through meeting,

a) The author or walk-through leader should make an overview presentation of the software product under examination
b) The walk-through leader shall coordinate a discussion of the general anomalies of concern
c) The author or walk-through leader shall present the software product, describing every portion of it
d) Team members shall raise specific anomalies as the author reaches the part of the software product to which the anomalies relate
e) The recorder shall note recommendations and actions arising out of the discussion upon each anomaly

After the walk-through meeting, the walk-through leader shall issue the walk-through output detailing anomalies, decisions, actions, and other information of interest. Minimum content requirements for the walk-through output are provided in 7.7.

7.5.6 Rework/follow-up

The walk-through leader shall verify that the action items assigned in the meeting are closed.

7.6 Exit criteria

The walk-through shall be considered complete when

a) The entire software product has been examined
b) Recommendations and required actions have been recorded
c) The walk-through output has been completed

7.7 Output

The output of the walk-through shall be documented evidence that identifies

a) The walk-through team members
b) The software product being examined
c) The statement of objectives that were to be accomplished during this walk-through meeting and whether they were met
d) A list of the recommendations made regarding each anomaly
e) A list of actions, due dates, and responsible people
f) Any recommendations made by the walk-through team on how to dispose of deficiencies and unresolved anomalies
g) Any proposals made by the walk-through team for follow-up walk-throughs

Although this standard sets minimum requirements for the content of the documented evidence, it is left to local procedures to prescribe additional content, format requirements, and media.

7.8 Data collection recommendations

Walk-throughs should provide data for the analysis of the quality of the software product, the effectiveness of the acquisition, supply, development, operation, and maintenance processes, and the efficiency of the walk-through itself. In order to maintain the effectiveness of walk-throughs, data should not be used to evaluate the performance of individuals. To enable these analyses, anomalies that are identified at a walk-through meeting should be classified in accordance with 7.8.1 through 7.8.3.

Walk-through data should contain the identification of the software product, the date and time of the walk-through, the walk-through leader, the preparation and walk-through times, the volume of the materials walked through, and the disposition of the software product. The capture of this information can be used to optimize local guidance for walk-throughs.

The management of walk-through data requires a capability to store, enter, access, update, summarize, and report categorized anomalies. The frequency and types of the walk-through analysis reports, and their distribution, are left to local standards and procedures.

7.8.1 Anomaly classification

Anomalies may be classified by technical type according to, for example, IEEE Std 1044-1993 [B7].

7.8.2 Anomaly classes

Anomaly classes provide evidence of nonconformance, and may be categorized, for example, as

a) Missing
b) Extra (superfluous)
c) Ambiguous
d) Inconsistent
e) Improvement desirable
f) Not conforming to standards
g) Risk-prone, i.e., the review finds that although an item was not shown to be “wrong,” the approach taken involves risks (and there are known safer alternative methods)
h) Factually incorrect
i) Not implementable (e.g., because of system constraints or time constraints)
j) Editorial

7.8.3 Anomaly ranking

Anomalies may be ranked by potential impact on the software product, for example, as

a) *Major.* Anomalies that would result in failure of the software product or an observable departure from specification
b) *Minor.* Anomalies that deviate from relevant specifications but will not cause failure of the software product or an observable departure in performance

7.9 Improvement

Walk-through data should be analyzed regularly in order to improve the walk-through itself and to improve the software activities used to produce the software product. Frequently occurring anomalies may be included in the walk-through checklists or role assignments. The checklists themselves should also be inspected regularly for superfluous or misleading questions. The preparation times, meeting times, and num-

ber of participants should be analyzed to determine connections between preparation rate, meeting rate, and number and severity of anomalies found.

8. Audits

8.1 Introduction

The purpose of a software audit is to provide an independent evaluation of conformance of software products and processes to applicable regulations, standards, guidelines, plans, and procedures.

Examples of software products subject to audit include, but are not limited to, the following:

a) Back-up and recovery plans
b) Contingency plans
c) Contracts
d) Customer or user representative complaints
e) Disaster plans
f) Hardware performance plans
g) Installation plans
h) Installation procedures
i) Maintenance plans
j) Management review reports
k) Operations and user manuals
l) Procurement and contracting methods
m) Reports and data (for example, review, audit, project status, anomaly reports, test data)
n) Request for proposal
o) Risk management plans
p) Software configuration management plans (see IEEE Std 828-1990 [B2])
q) Software design descriptions (see IEEE Std 1016-1987 [B6])
r) Source code
s) Unit development folders
t) Software project management plans (see IEEE Std. 1058-1987 [B8])
u) Software quality assurance plans (see IEEE Std 730-1989 [B1])
v) Software requirements specifications (see IEEE Std 830-1993 [B4])
w) Software safety plans (see IEEE Std 1228-1994 [B13])
x) Software test documentation (see IEEE Std 829-1983 [B3])
y) Software user documentation (see IEEE Std 1063-1987 [B9])
z) Software verification and validation plans (see IEEE Std 1012-1986 [B5])
aa) Standards, regulations, guidelines, and procedures
ab) System build procedures
ac) Technical review reports
ad) Vendor documents
ae) Walk-through reports
af) Deliverable media (such as tapes and diskettes)

The examination should begin with an overview meeting during which the auditors and audited organization examine and agree upon the arrangements for the audit.

When stipulated in the audit plan, the auditors may make recommendations. These should be reported separately.

8.2 Responsibilities

The following roles shall be established for an audit:

a) Lead auditor
b) Recorder
c) Auditor(s)
d) Initiator
e) Audited organization

The lead auditor may act as recorder. The initiator may act as lead auditor. Additional auditors should be included in the audit team; however, audits by a single person are permitted.

8.2.1 Lead auditor

The lead auditor shall be responsible for the audit. This responsibility includes administrative tasks pertaining to the audit, ensuring that the audit is conducted in an orderly manner, and ensuring that the audit meets its objectives.

a) Preparing the audit plan (see 8.5.2)
b) Assembling the audit team
c) Managing the audit team
d) Making decisions regarding the conduct of the audit
e) Making decisions regarding any audit observations
f) Preparing the audit report (see 8.7)
g) Reporting on the inability or apparent inability of any of individuals involved in the audit to fulfill their responsibilities
h) Negotiating any discrepancies or inconsistencies with the initiator which could impair the ability to satisfy the exit criteria (8.6)
i) Recommending corrective actions

The lead auditor shall be free from bias and influence that could reduce his ability to make independent, objective evaluations.

8.2.2 Recorder

The recorder shall document anomalies, action items, decisions, and recommendations made by the audit team.

8.2.3 Auditor

The auditors shall examine products, as defined in the audit plan. They shall document their observations and recommend corrective actions. All auditors shall be free from bias and influences that could reduce their ability to make independent, objective evaluations, or shall identify their bias and proceed with acceptance from the initiator.

8.2.4 Initiator

The initiator shall be responsible for the following activities:

a) Decide upon the need for an audit
b) Decide upon the purpose and scope of the audit
c) Decide the software products to be audited

d) Decide the evaluation criteria, including the regulations, standards, guidelines, plans, and procedures to be used for evaluation
e) Decide upon who will carry out the audit
f) Review the audit report
g) Decide what follow-up action will be required
h) Distribute the audit report

The initiator may be a manager in the audited organization, a customer or user representative of the audited organization, or a third party.

8.2.5 Audited organization

The audited organization shall provide a liaison to the auditors and shall provide all information requested by the auditors. When the audit is completed, the audited organization should implement corrective actions and recommendations.

8.3 Input

Inputs to the audit shall be listed in the audit plan and shall include the following:

a) Purpose and scope of the audit
b) Background information about the audited organization
c) Software products to be audited
d) Evaluation criteria, including applicable regulations, standards, guidelines, plans, and procedures to be used for evaluation
e) Evaluation criteria: for example, “acceptable,” “needs improvement,” “unacceptable,” “not rated”

Inputs to the audit should also include the following:

f) Records of previous similar audits

8.4 Entry criteria

8.4.1 Authorization

An initiator decides upon the need for an audit. This decision may be prompted by a routine event, such as the arrival at a project milestone, or a non-routine event, such as the suspicion or discovery of a major non-conformance.

The initiator selects an auditing organization that can perform an independent evaluation. The initiator provides the auditors with information that defines the purpose of the audit, the software products to be audited, and the evaluation criteria. The initiator should request the auditors to make recommendations. The lead auditor produces an audit plan and the auditors prepare for the audit.

The need for an audit may be established by one or more of the following events:

a) The supplier organization decides to verify compliance with the applicable regulations, standards, guidelines, plans, and procedures (this decision may have been made when planning the project).
b) The customer organization decides to verify compliance with applicable regulations, standards, guidelines, plans, and procedures.
c) A third party, such as a regulatory agency or assessment body, decides upon the need to audit the supplier organization to verify compliance with applicable regulations, standards, guidelines, plans, and procedures.

In every case, the initiator shall authorize the audit.

8.4.2 Preconditions

An audit shall be conducted only when all of the following conditions have been met:

a) The audit has been authorized by an appropriate authority
b) A statement of objectives of the audit is established
c) The required audit inputs are available

8.5 Procedures

8.5.1 Management preparation

Managers shall ensure that the audit is performed as required by applicable standards and procedures and by requirements mandated by law, contract, or other policy. To this end, managers shall

a) Plan time and resources required for audits, including support functions, as required in IEEE Std 1058.1-1987 [B8], legal or regulatory documents, or other appropriate standards
b) Provide funding and facilities required to plan, define, execute, and manage the audits
c) Provide training and orientation on the audit procedures applicable to a given project
d) Ensure appropriate levels of expertise and knowledge sufficient to comprehend the software product being audited
e) Ensure that planned audits are conducted
f) Act on audit team recommendations in a timely manner

8.5.2 Planning the audit

The audit plan shall describe the

a) Purpose and scope of the audit
b) Audited organization, including location and management
c) Software products to be audited
d) Evaluation criteria, including applicable regulations, standards, guidelines, plans, and procedures to be used for evaluation
e) Auditor’s responsibilities
f) Examination activities (for example, interview staff, read and evaluate documents, observe tests)
g) Audit activity resource requirements
h) Audit activity schedule
i) Requirements for confidentiality (for example, company confidential, restricted information, classified information)
j) Checklists
k) Report formats
l) Report distribution
m) Required follow-up activities

Where sampling is used, a statistically valid sampling method shall be used to establish selection criteria and sample size.

The audit plan shall be approved by the initiator. The audit plan should allow for changes based on information gathered during the audit, subject to approval by the initiator.

8.5.3 Opening meeting

An opening meeting between the audit team and audited organization shall occur at the beginning of the examination phase of the audit. The overview meeting agenda shall include

a) Purpose and scope of the audit
b) Software products being audited
c) Audit procedures and outputs
d) Expected contributions of the audited organization to the audit (for example, the number of people to be interviewed, meeting facilities)
e) Audit schedule
f) Access to facilities, information, and documents required

8.5.4 Preparation

The initiator shall notify the audited organization's management in writing before the audit is performed, except for unannounced audits. The notification shall define the purpose and scope of the audit, identify what will be audited, identify the auditors, and identify the audit schedule. The purpose of notification is to enable the audited organization to ensure that the people and material to be examined in the audit are available.

Auditors shall prepare for the audit by studying the

a) Audit plan
b) Audited organization
c) Products to be audited
d) Applicable regulations, standards, guidelines, plans, and procedures to be used for evaluation
e) Evaluation criteria

In addition, the lead auditor shall make the necessary arrangements for

f) Team orientation and training
g) Facilities for audit interviews
h) Materials, documents, and tools required by the audit procedures
i) Examination activities

8.5.5 Examination

Examination shall consist of evidence collection and analysis with respect to the audit criteria, a closing meeting between the auditors and audited organization, and preparing an audit report.

8.5.5.1 Evidence collection

The auditors shall collect evidence of conformance and non-conformance by interviewing audited organization staff, examining documents, and witnessing processes. The auditors should attempt all the examination activities defined in the audit plan. They shall undertake additional investigative activities if they consider such activities required to define the full extent of conformance or non-conformance.

Auditors shall document all observations of non-conformance and exemplary conformance. An observation is a statement of fact made during an audit that is substantiated by objective evidence. Examples of non-conformance are

a) Applicable regulations, standards, guidelines, plans, and procedures not used at all
b) Applicable regulations, standards, guidelines, plans, and procedures not used correctly

Observations should be categorized as major or minor. An observation should be classified as major if the non-conformity will likely have a significant effect on product quality, project cost, or project schedule.

All observations shall be verified by discussing them with the audited organization before the closing audit meeting.

8.5.5.2 Closing meeting

The lead auditor shall convene a closing meeting with the audited organization's management. The closing meeting should review

a) Actual extent of implementation of the audit plan
b) Problems experienced in implementing the audit plan, if any
c) Observations made by the auditors
d) Preliminary conclusions of the auditors
e) Preliminary recommendations of the auditors
f) Overall audit assessment (for example, whether the audited organization successfully passed the audit criteria)

Comments and issues raised by the audited organization should be resolved. Agreements should be reached during the closing audit meeting and must be completed before the audit report is finalized.

8.5.5.3 Reporting

The lead auditor shall prepare the audit report, as described in 8.7. The audit report should be prepared as soon as possible after the audit. Any communication between auditors and the audited organization made between the closing meeting and the issue of the report should pass through the lead auditor.

The lead auditor shall send the audit report to the initiator. The initiator should distribute the audit report within the audited organization.

8.5.6 Follow-up

Rework, if any, shall be the responsibility of the initiator and audited organization and shall include

a) Determining what corrective action is required to remove or prevent a non-conformity
b) Initiating the corrective action

8.6 Exit criteria

An audit shall be considered complete when

a) The audit report has been submitted to the initiator
b) All of the auditing organization's follow-up actions included in the scope of the audit have been performed, reviewed, and approved

8.7 Output

The output of the audit is the audit report. The audit report shall contain the

a) Purpose and scope of the audit
b) Audited organization, including location, liaison staff, and management
c) Identification of the software products audited

d) Applicable regulations, standards, guidelines, plans, and procedures used for evaluation
e) Evaluation criteria
f) Summary of auditor's organization
g) Summary of examination activities
h) Summary of the planned examination activities not performed
i) Observation list, classified as major or minor
j) A summary and interpretation of the audit findings including the key items of non-conformance
k) The type and timing of audit follow-up activities

Additionally, when stipulated by the audit plan, recommendations shall be provided to the audited organization or the initiator. Recommendations may be reported separately from results.

Although this standard sets minimum requirements for report content, it is left to local standards to prescribe additional content, report format requirements, and media.

Annex A

(informative)

Relationship of this standard to the life cycle processes of other standards

This standard may be used in conjunction with other IEEE or ISO/IEC standards. In particular, IEEE Std 730-1989 [B1], IEEE Std 1012-1986 [B5], IEEE Std 1074-1995 [B10], and ISO/IEC 12207:1995 [B15] all require that software reviews take place during the software life cycle. The following table shows, for each of these standards, a possible mapping to the five review types described in the body of IEEE Std 1028-1997.

Standard	Clause	Review title	Corresponding IEEE Std 1028-1997 review type
IEEE Std 730-1989 [B1]	3.6.2.1	Software requirements review	Technical review
	3.6.2.2	Preliminary design review	Technical review
	3.6.2.3	Critical design review	Technical review
	3.6.2.4	Software V& V plan review	Management review
	3.6.2.5	Functional audit	Audit
	3.6.2.6	Physical audit	Audit
	3.6.2.7	In-process audit	Audit
	3.6.2.8	Managerial reviews	Management review
	3.6.2.9	Software configuration management revIew	Management review
	3.6.2.10	Postmortem review	Management review, technical review
IEEE Std 1012-1986 [B5]	3.5.2	Concept documentation evaluation	Technical review
	3.5.3	Software requirements traceability analysis, requirements evaluation, and interface analysis	Technical review, inspection, walk-through
	3.5.4	Design traceability analysis, design evaluation, and design interface analysis	Technical review, inspection, walk-through
	3.5.5	Source code traceability analysis, evaluation, and interface analysis	Technical review, inspection, walk-through
	3.5.5	Source code documentation evaluation	Technical review, inspection, walk-through

Standard	Clause	Review title	Corresponding IEEE Std 1028-1997 review type
IEEE Std 1012-1986 [B5]	Appendix	Algorithm analysis	Technical review, inspection, walk-through
		Audit performance	Audit
		Configuration control audit	Audit
		Control flow analysis	Technical review, walk-through
		Database analysis	Technical review, inspection, walk-through
		Data flow analysis	Technical review, inspection, walk-through
		Design walk-through	Walk-through
		Feasibility study evaluation	Management review
		Functional audit	Audit
		In-process audit	Audit
		Operational readiness review	Management review, technical review
		Physical audit	Audit
		Requirements walk-through	Walk-through
		Sizing and timing analysis	Technical review, inspection, walk-through
		Source code walk-through	Walk-through
		Test evaluation	Technical review, inspection, audit
		Test readiness review	Management review, technical review
		Test walk-through	Walk-through
		User documentation evaluation	Technical review, audit
IEEE Std 1074-1995 [B10]	7.1.3.2	Plan verification and validation	All types
ISO/IEC 12207:1995 [B15]	5.2.4.5	Project management plan	Management review, technical review, inspection, walk-through, audit
	5.2.6.2	Supplier/acquirer joint reviews	Management review, technical review, inspection, walk-through, audit

Standard	Clause	Review title	Corresponding IEEE Std 1028-1997 review type
ISO/IEC 12207:1995 [B15]	5.3.1.3	Development process	Management review, technical review, inspection, walk-through, audit
	5.3.2.2	System requirements analysis evaluation	Technical review, inspection, walk-through
	5.3.3.2	System architectural design evaluation	Technical review, inspection, walk-through
	5.3.4.2	Software requirements analysis evaluation	Technical review, inspection, walk-through
	5.3.5.6	Software architectural design evaluation	Technical review, inspection, walk-through
	5.3.6.7	Software detailed design evaluation	Technical review, inspection, walk-through
	5.3.7.5	Software code evaluation	Technical review, inspection, walk-through
	5.3.8.5	Software integration evaluation	Technical review, inspection, walk-through
	5.3.9.3	Software qualification testing evaluation	Technical review, inspection, walk-through
	5.3.10.3	System integration evaluation	Technical review, inspection, walk-through
	5.3.11.2	System qualification test evaluation	Technical review, inspection, walk-through
	6.1.2.3	Document review	Management review, technical review, inspection, walk-through, audit
	6.6.2	Project management reviews	Management review
	6.6.3	Technical reviews	Technical review
	6.7	Audit process	Audit
	7.1.4	Review and evaluation	Management review, technical review
	B.3.c	Tailoring—reviews and audits	Inspection, walk-through

Annex B

(informative)

Comparison of review types

The following table compares the five types of reviews in a number of salient characteristics. This is meant to be indicative of the ways in which the review types match with or differ from one another.

Characteristic	Management review	Technical review	Inspection	Walk-through	Audit
Objective	Ensure progress; recommend corrective action; ensure proper allocation of resources	Evaluate conformance to specifications and plans; ensure change integrity	Find anomalies; verify resolution; verify product quality	Find anomalies; examine alternatives; improve product; forum for learning	Independently evaluate compliance with objective standards and regulations
Decision-making	Management team charts course of action; decisions made at the meeting or as a result of recommendations	Review team requests management or technical leadership to act on recommendations	Review team chooses predefined product dispositions; defects must be removed	The team agrees on changes to be made by the author	Audited organization, initiator, acquirer, customer or user
Change verification	Leader verifies that action items are closed; change verification left to other project controls	Leader verifies that action items are closed; change verification left to other project controls	Leader verifies that action items are closed; change verification left to other project controls	Leader verifies that action items are closed; change verification left to other project controls	Responsibility of the audited organization
Recommended group size	Two or more people	Three or more people	Three to six people	Two to seven people	One to five people
Group attendance	Management, technical leadership and peer mix	Technical leadership and peer mix	Peers meet with documented attendance	Technical leadership and peer mix	Auditors, audited organization, management and technical personnel
Group leadership	Usually the responsible manager	Usually the lead engineer	Trained facilitator	Facilitator or author	Lead auditor
Volume of material	Moderate to high, depending on the specific meeting objectives	Moderate to high, depending on the specific meeting objectives	Relatively low	Relatively low	Moderate to high, depending on the specific audit objectives

Characteristic	Management review	Technical review	Inspection	Walk-through	Audit
Presenter	Project representative	Development team representative	A reader	Author	Auditors collect and examine information provided by audited organization
Data collection	As required by applicable policies, standards, or plans	Not a formal project requirement. May be done locally.	Strongly recommended	Recommended	Not a formal project requirement. May be done locally.
Output	Management review documentation	Technical review documentation	Anomaly list, anomaly summary, inspection documentation	Anomaly list, action items, decisions, follow-up proposals	Formal audit report; observations, findings, deficiencies
Formal facilitator training	No	No	Yes	No	Yes (formal auditing training)
Defined participant roles	Yes	Yes	Yes	Yes	Yes
Use of defect checklists	No	No	Yes	No	Yes
Management participates	Yes	Optional	No	No	Yes
Customer or user representative participates	Optional	Optional	Optional	Optional	Optional

Annex C

(informative)

Bibliography

The standards listed here may be useful in the preparation of software products that can be reviewed using the procedure documented in this standard:

[B1] IEEE Std 730-1989, IEEE Standard for Software Quality Assurance Plans.[3]

[B2] IEEE Std 828-1990, IEEE Standard for Software Configuration Management Plans.

[B3] IEEE Std 829-1983 (R1991), IEEE Standard for Software Test Documentation.

[B4] IEEE Std 830-1993, IEEE Recommended Practice for Software Requirements Specifications.

[B5] IEEE Std 1012-1986 (R1992), IEEE Standard for Software Verification and Validation Plans.

[B6] IEEE Std 1016-1987 (R1993), IEEE Recommended Practice for Software Design Descriptions.

[B7] IEEE Std 1044-1993, IEEE Standard Classification for Software Anomalies.

[B8] IEEE Std 1058-1987 (R1993), IEEE Standard for Software Project Management Plans.

[B9] IEEE Std 1063-1987 (R1993), IEEE Standard for Software User Documentation.

[B10] IEEE Std 1074-1995, IEEE Standard for Developing Software Life Cycle Processes.

[B11] IEEE Std 1219-1992, IEEE Standard for Software Maintenance.

[B12] IEEE Std 1220-1994, IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process.

[B13] IEEE Std 1228-1994, IEEE Standard for Software Safety Plans.

[B14] IEEE Std 1298-1992 (AS 3563.1-1991), IEEE Standard for Software Quality Management System, Part 1: Requirements.

[B15] ISO/IEC 12207:1995, Information technology—Software life cycle processes.[4]

[B16] ISO 9001:1994, Quality systems—Model for quality assurance in design/development, production, installation and servicing.

[B17] ISO 10011-1:1990, Guidelines for auditing quality systems—Part 1: Auditing.

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

[4]ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, 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.