730™

IEEE Standard for Software Quality Assurance Plans

IEEE Computer Society

Sponsored by the
Software Engineering Standards Committee

IEEE

IEEE Standard for Software Quality Assurance Plans

Sponsor

Software Engineering Standards Committee
of the
IEEE Computer Society

Approved 12 September 2002

IEEE-SA Standards Board

Abstract: The standard specifies the format and content of software quality assurance plans. It meets the IEEE/EIA 12207.1 requirements for such plans.
Keywords: assurance, quality, software quality

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

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.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

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

> Secretary, IEEE-SA 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 patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

This standard assists in determining the content and preparation of Software Quality Assurance Plans and provides a standard against which such plans can be prepared and assessed. It is directed toward the development and maintenance of software.

This standard is consistent with Clause 6.20 of IEEE/EIA 12207.1-1997 [B18] and obviates the need for Annex A of the 1998 version of the standard.[1] In addition to eliminating Annex A of the 1998 version of this standard, minor structural changes have been made to conform to the current IEEE Standards Style Manual.

There are three groups to whom this standard applies or may affect: the user, the supplier, and the public.

a) The user, who may be another element of the same organization developing the software, has a need for the product. Further, the user needs the product to meet the requirements identified in the specification. The user thus cannot afford a "hands-off" attitude toward the supplier and rely solely on a test to be executed at the end of the software development time period. If the product should fail, not only does the same need still exist, but also a portion of the development time has been lost. Therefore, the user needs to obtain a reasonable degree of confidence that the product is in the process of acquiring required attributes during software development.

b) The supplier needs an established standard against which to plan and to be measured. It is unreasonable to expect a complete reorientation from project to project. Not only is it not cost effective, but, unless there exists a stable framework on which to base changes, improvement cannot be made.

c) The public may be affected by the use of the product. The public may include, for example, depositors at a bank or passengers using a reservation system. The public users have requirements, such as legal rights, which preclude haphazard development of software. At some later date, the provider of the services to the public user and the supplier may be required to show that they acted in a reasonable and prudent professional manner to ensure that required software attributes were acquired.

Participants

This standard was prepared by the Software Engineering Standards Committee of the Software Engineering Technical Committee of the IEEE Computer Society. The Working Group consisted of the following members:

R. Shillato, *Chair*

R. Dean   J. Horch   D. Schultz

---

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

The following members of the balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Edward Addy	John Garth Glynn	Lou Pinto
Franz Philippe Bachmann	Lawrence Gunther	Garry Roedler
Juris Borzovs	Herbert Hecht	Terence Rout
Christine Brown-Strysik	Mark Heinrich	Jaideep Roy
Dino Butorac	John Horch	James Ruggieri
Susan Carroll	William Junk	James Sanders
Muralikrishna Chemuturi	George Kambic	Hans Schaefer
Antonio Cicu	Dwayne Knirk	David Schultz
Rosemary Coleman	Sunil Kumar	Robert W. Shillato
Paul Croll	Thomas M. Kurihara	Mitchell Smith
Guru Dutt Dhingra	J. Dennis Lawrence	Joyce Statz
Gregory Daich	Jacques Mathot	Toru Taqkeshita
Taz Daughtrey	Ian McChesney	Richard H. Thayer
Bostjan Derganc	Denis Meredith	Scott Valcourt
Christof Ebert	Jerome Mersky	Richard Walker
Caroline Evans	James Moore	John Walz
William Eventoff	Robert Mortonson	John Williams
David Franklin	Dennis Nickle	Paul Wolfgang
Juan Garbajosa	Susumu Ohno	Oren Yuen
Barry Garner	Gerry Ourada	Janusz Zalewski
Gregg Giesler		Geraldine Zimmerman

When the IEEE-SA Standards Board approved this standard on 12 September 2002, it had the following membership:

James T. Carlo, *Chair*
James H. Gurney, *Vice Chair*
Judith Gorman, *Secretary*

Sid Bennett	Toshio Fukuda	Nader Mehravari
H. Stephen Berger	Arnold M. Greenspan	Daleep C. Mohla
Clyde R. Camp	Raymond Hapeman	William J. Moylan
Richard DeBlasio	Donald M. Heirman	Malcolm V. Thaden
Harold E. Epstein	Richard H. Hulett	Geoffrey O. Thompson
Julian Forster*	Lowell G. Johnson	Howard L. Wolfman
Howard M. Frazier	Joseph L. Koepfinger*	Don Wright
	Peter H. Lips	

*Member Emeritus

Also included is the following nonvoting IEEE-SA Standards Board liaison:

Alan Cookson, *NIST Representative*
Satish K. Aggarwal, *NRC Representative*

Noelle D. Humenick
IEEE Standards Project Editor

Contents

IEEE Standard for Software Quality Assurance Plans

1. Overview

1.1 Scope

This standard applies to the development of a software quality assurance plan (SQAP). The existence of this standard should not be construed to prohibit additional content in a SQAP. An assessment should be made for the specific software item to assure adequacy of coverage. Where this standard is invoked for an organization or project engaged in producing several software items, the applicability of the standard should be specified for each of the software items.

Although this document does not require the use of IEEE/EIA Std 12207.0-1996 [B17] and IEEE/EIA Std 12207.1-1997 [B18], it is consistent with those two standards.[1] An SQAP meeting the requirements of this standard will be in document compliance with the SQAP information item of IEEE/EIA 12207.1-1997 [B18].

1.2 Purpose

The purpose of this standard is to provide uniform, minimum acceptable requirements for preparation and content of software quality assurance plans.

In considering adoption of this standard, regulatory bodies should be aware that specific application of this standard may already be covered by one or more IEEE or ANSI standards documents relating to quality assurance, definitions, or other matters. It is not the purpose of this document to supersede, revise, or amend existing standards directed to specific industries or applications.

1.3 Conformance to this standard

Content conformance to this standard can be claimed when all requirements (indicated by "shall") are carried out as defined in this standard. Format conformance to this standard can be claimed if the resulting SQAP is in the format specified in Clause 4. 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] The numbers in brackets correspond to those of the bibliography in Annex A.

1

2. References

Not applicable. Annex A contains bibliographic information that may be useful in the preparation of SQAPs.

3. Definitions and acronyms

3.1 Definitions

Definitions of terms can be found in *The Authoritative Dictionary of IEEE Standards Terms,* Seventh Edition [B1] and IEEE Std 610.12™-1990 [B2], or latest revision thereof. For the purpose of this standard, the term "software" includes firmware, documentation, data, and execution control statements (e.g., command files, job control language, etc.).

3.2 Acronyms

The following acronyms appear within the text of this standard:

ADR	architecture design review
DDR	detailed design review
SCM	software configuration management
SCMP	software configuration management plan
SCMPR	software configuration management plan review
SDD	software design description
SQA	software quality assurance
SQAP	software quality assurance plan
SRD	software requirements description
SSR	software specifications review
UDR	user documentation review

4. Software quality assurance plan

The SQAP shall include the sections listed below to be in format conformance with this standard. The sections shall be ordered in the described sequence or, if the sections are not ordered in the described sequence, a table shall be provided at the end of the SQAP that provides a cross-reference from the lowest numbered sub-section of the standard to that portion of the SQAP where the material is provided. If there is no information pertinent to a section, the following shall appear below the section heading, "This section is not applicable to this plan," together with the appropriate reasons for the exclusion.

The first page of the SQAP shall include the date of issue, the status of the document, and the identification of the issuing organization. The SQAP sections shall include the following:

1) Purpose
2) Reference documents
3) Management
4) Documentation

5) Standards, practices, conventions, and metrics
6) Software reviews
7) Test
8) Problem reporting and corrective action
9) Tools, techniques, and methodologies
10) Media control
11) Supplier control
12) Records collection, maintenance, and retention
13) Training
14) Risk management
15) Glossary
16) SQAP change procedure and history

Additional sections may be added as required.

Some of the material may appear in other documents. If so, then reference to these documents shall be made in the body of the SQAP. For those projects in which all required SQAP material is included in the project plan, or other governing document, the governing document shall map the material to the list of SQAP sections.

The SQAP shall be approved by the manager of each unit of the organization having responsibilities defined within this SQAP or their designated representatives.

Details for each section of the SQAP are described in 4.1 through 4.16 of this standard.

4.1 Purpose (section 1 of the SQAP)

This section shall delineate the specific purpose and scope of the particular SQAP. It shall list the name(s) of the software items covered by the SQAP and the intended use of the software. It shall state the portion of the software life cycle covered by the SQAP for each software item specified.

4.2 Reference documents (section 2 of the SQAP)

This section shall provide a complete list of documents referenced elsewhere in the text of the SQAP. Included in this list shall be the documents that were used in developing the SQAP, including policies or laws that give rise to the need for this plan as well as other plans or task descriptions that elaborate details of this plan. The version and date of each document shall be included in the list.

4.3 Management (section 3 of the SQAP)

This section shall describe the project's organization structure, its tasks, and its roles and responsibilities (see IEEE Std 1058™-1998 [B13]).

4.3.1 Organization

This section shall depict the organizational structure that influences and controls the quality of the software. This shall include a description of each major element of the organization together with the roles and delegated responsibilities. The amount of organizational freedom and objectivity to evaluate and monitor the quality of the software, and to verify problem resolutions, shall be clearly described and documented. In addition, the organization responsible for preparing and maintaining the SQAP shall be identified.

4.3.2 Tasks

This section shall describe:

a) That portion of the software life cycle covered by the SQAP.
b) The tasks to be performed.
c) The entry and exit criteria for each task.
d) The relationships between these tasks and the planned major checkpoints. The sequence and relationships of tasks, and their relationship to the project management plan master schedule, shall be indicated.

4.3.3 Roles and responsibilities

This section shall identify the specific organizational element that is responsible for performing each task.

4.3.4 Quality assurance estimated resources

This section shall provide the estimate of resources and the costs to be expended on quality assurance and quality control tasks.

4.4 Documentation (section 4 of the SQAP)

4.4.1 Purpose

This section shall perform the following functions:

a) Identify the documentation governing the development, verification and validation, use, and maintenance of the software.
b) List which documents are to be reviewed or audited for adequacy. For each document listed, identify the reviews or audits to be conducted and the criteria by which adequacy is to be confirmed, with reference to section 6 of the SQAP.

4.4.2 Minimum documentation requirements

To ensure that the implementation of the software satisfies the technical requirements, the following documentation is required as a minimum.

NOTE—Names of various artifacts listed below use the terminology contained in IEEE/EIA 12207.1-1997 [B18]. The content of these documents may be included in other documents as long as traceability to the required information is maintained; e.g., a reference in that section of the document or a traceability table.

4.4.2.1 Software requirements description (SRD)

The SRD should specify requirements for a particular software product, program, or set of programs that perform certain functions in a specific environment. The SRD may be written by the supplier (internal or external), the customer, or by both. The SRD should address the basic issues of functionality, external interfaces, performance, attributes, and design constraints imposed on implementation. Each requirement should be uniquely identified and defined such that its achievement is capable of being objectively verified and validated (see IEEE Std 830™-1998 [B5]).

4.4.2.2 Software design description (SDD)

The SDD should depict how the software will be structured to satisfy the requirements in the SRD. The SDD should describe the components and subcomponents of the software design, including databases and internal

4

interfaces. The SDD may be prepared first as the architecture design (also sometimes referred to as the *top-level SDD*) and should be subsequently expanded to produce the detailed SDD (see IEEE Std 1016™-1998 [B11]).

4.4.2.3 Verification and validation plans

Verification and validation processes are used to determine if developed software products conform to their requirements, and whether the software products fulfill the intended use and user expectations. This includes analysis, evaluation, review, inspection, assessment, and testing of the software products and the processes that produced the products. Also, the software testing, validation, and verification processes apply when integrating purchased or customer-supplied software products into the developed product.

The verification plan should document the verification tasks and the validation plan should document the validation tasks. If desired, the verification plan and validation plan may be packaged together in a single document. Each plan defines the verification and validation tasks and required inputs and outputs needed to maintain the appropriate software integrity level. It also provides a means of verifying the implementation of the requirements of the SRD in the design as expressed in the SDD and in the testing as expressed in the project's test documentation (see IEEE Std 829™-1998 [B4], IEEE Std 1008™-1987 [B8], and IEEE Std 1012™-1998 [B9], and IEEE Std 1012a™-1998 [B10]).

4.4.2.4 Verification results report and validation results report

The verification results report should describe the results of the software verification activities conducted according to the verification plan. The validation results report should describe the results of the software test or validation carried out according to the validation plan.

4.4.2.5 User documentation

User documentation guides the users in installing, operating, managing, and maintaining (does not apply when modifying software source code) software products.

The user documentation should describe the data control inputs, input sequences, options, program limitations, and all other essential information for the software product. All error messages should be identified and described. All corrective actions to correct the errors causing the error messages shall be described. The documentation is applicable to any portion of the embedded software with which the user interacts directly. (see IEEE Std 1063™-1987 [B14]).

4.4.2.6 Software configuration management plan (SCMP)

The SCMP should document what software configuration management (SCM) activities should be accomplished, how they should be accomplished, who is responsible for specific tasks, the schedule of events, and what resources will be utilized. The SCMP should also define the methods and facilities used to maintain, store, secure, and document controlled versions and related artifacts of the identified software during all phases of the software life cycle. This may be implemented in conjunction with a computer program library. Part of the plan includes documenting how the release and delivery of the product is managed. The plan can address SCM tasks over any portion of the product's development life cycle. At a minimum, the plan should address the SCM tasks that apply to the portion of the life cycle covered by the SQAP (see IEEE Std 828™-1998 [B3]).

4.4.3 Other documentation

Identify other documents applicable to the software development project and software product that may be required. Other documentation may include the following:

NOTE—Names of various artifacts listed below use the terminology contained in IEEE/EIA 12207.1-1997 [B18].

   a)   Development process plan
   b)   Software development standards description
   c)   Software engineering methods/procedures/tools description
   d)   Software project management plan (see IEEE Std 1058™-1998 [B13])
   e)   Maintenance plan (see IEEE Std 1219™-1998 [B15])
   f)   Software safety plans (see IEEE Std 1228™ -1994 [B16])
   g)   Software integration plan

4.5 Standards, practices, conventions, and metrics (section 5 of the SQAP)

4.5.1 Purpose

This section shall:

   a)   Identify the standards, practices, conventions, statistical techniques to be used, quality requirements, and metrics to be applied. Product and process measures shall be included in the metrics used and may be identified in a separate measurement plan (see IEEE Std 1219-1998 [B15] and IEEE Std 1228-1994 [B16]).
   b)   State how conformance with these items is to be monitored and assured.

4.5.2 Content

The subjects covered shall include the basic technical, design, and programming activities involved, such as documentation, variable and module naming, programming, inspection, and testing. As a minimum, the following information shall be provided (see IEEE Std 982.1™-1988 [B6] and IEEE Std 982.2™-1988 [B7]):

   a)   Documentation standards
   b)   Design standards
   c)   Coding standards
   d)   Commentary standards
   e)   Testing standards and practices
   f)   Selected software quality assurance product and process metrics

4.6 Software reviews (section 6 of the SQAP)

4.6.1 Purpose

This section shall (see IEEE Std 1028™-1997 [B12]):

   a)   Define the software reviews to be conducted. They may include managerial reviews, acquirer-supplier reviews, technical reviews, inspections, walk-throughs, and audits.
   b)   List the schedule for software reviews as they relate to the software project's schedule.
   c)   State how the software reviews shall be accomplished.
   d)   State what further actions shall be required and how they shall be implemented and verified.

4.6.2 Minimum requirements

As a minimum, the following software reviews shall be conducted.

4.6.2.1 Software specifications review (SSR)

The SSR is held to assure the adequacy of the requirements stated in the SRD.

4.6.2.2 Architecture design review (ADR)

The ADR is held to evaluate the technical adequacy of the preliminary design (also known as *top-level design*) of the software as depicted in the preliminary software design description.

4.6.2.3 Detailed design review (DDR)

The DDR is held to determine the acceptability of the detailed software designs as depicted in the detailed software design description in satisfying the requirements of the SRD.

4.6.2.4 Verification and validation plan review

The verification and validation plan review is held to evaluate the adequacy and completeness of the verification and validation methods defined in the verification and validation plans.

4.6.2.5 Functional audit

This audit is held prior to the software delivery to verify that all requirements specified in the SRD have been met.

4.6.2.6 Physical audit

This audit is held to verify internal consistency of the software and its documentation, and their readiness for release.

4.6.2.7 In-process audits

In-process audits of samples of the design are held to verify the consistency of the design, including:

a) Code versus design documentation
b) Interface specifications (hardware and software)
c) Design implementations versus functional requirements
d) Functional requirements versus test descriptions

4.6.2.8 Managerial reviews

Managerial reviews are held periodically to assess the execution of all of the actions and the items identified in the SQAP. Management reviews are carried out by, or on behalf of, the management personnel having direct responsibility for the system. This review may require additional changes in the SQAP itself.

4.6.2.9 Software configuration management plan review (SCMPR)

The SCMPR is held to evaluate the adequacy and completeness of the configuration management methods defined in the SCMP.

4.6.2.10 Post-implementation review

This review is held at the conclusion of the project to assess the development activities on that project and to provide recommendations for appropriate actions.

4.6.3 Other reviews and audits

Other reviews and audits may include the user documentation review (UDR). This review is held to evaluate the adequacy (e.g., completeness, clarity, correctness, and usability) of the user documentation.

4.7 Test (section 7 of the SQAP)

This section shall identify all the tests not included in the software verification and validation plan for the software covered by the SQAP and shall state the methods to be used. If a separate test plan exists it shall be referenced.

4.8 Problem reporting and corrective action (section 8 of the SQAP)

This section shall:

a)   Describe the practices and procedures to be followed for reporting, tracking, and resolving problems or issues identified in both software items and the software development and maintenance process.
b)   State the specific organizational responsibilities concerned with their implementation.

4.9 Tools, techniques, and methodologies (section 9 of the SQAP)

This section shall identify the software tools, techniques, and methods used to support SQA processes. For each, this section shall state the intended use, applicability, or circumstances under which it is to be used or not to be used, and limitations.

4.10 Media control (section 10 of the SQAP)

This section shall state the methods and facilities to be used to

a)   Identify the media for each intermediate and deliverable computer work product and the documentation required to store the media, including the copy and restore process.
b)   Protect computer program physical media from unauthorized access or inadvertent damage or degradation during all phases of the software life cycle. This may be provided as a part of the SCMP. If so, an appropriate reference shall be made thereto.

4.11 Supplier control (section 11 of the SQAP)

This section shall state the provisions for assuring that software provided by suppliers meets established requirements. In addition, this section shall state the methods that will be used to assure that the software supplier receives adequate and complete requirements. For previously developed software, this section shall state the methods to be used to ensure the suitability of the product for use with the software items covered by the SQAP. For software that is to be developed, the supplier shall be required to prepare and implement an SQAP in accordance with this standard. This section shall also state the methods to be employed to assure that the suppliers comply with the requirements of this standard. If software is to be developed under contract, then the procedures for contract review and update shall be described.

4.12 Records collection, maintenance, and retention (section 12 of the SQAP)

This section shall identify the SQA documentation to be retained, shall state the methods and facilities to be used to assemble, file, safeguard, and maintain this documentation, and shall designate the retention period.

4.13 Training (section 13 of the SQAP)

This section shall identify the training activities necessary to meet the needs of the SQAP.

4.14 Risk management (section 14 of the SQAP)

This section shall specify the methods and procedures employed to identify, assess, monitor, and control areas of risk arising during the portion of the software life cycle covered by the SQAP.

4.15 Glossary (section 15 of the SQAP)

This section shall contain a glossary of terms unique to the SQAP.

4.16 SQAP change procedure and history (section 16 of the SQAP)

This section shall contain the procedures for modifying the SQAP and maintaining a history of the changes. It shall also contain a history of such modifications.

Annex A

(informative)

Bibliography

[B1] IEEE 100™, *The Authoritative Dictionary of IEEE Standards Terms,* Seventh Edition.[2]

[B2] IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology

[B3] IEEE Std 828-1998, IEEE Standard for Software Configuration Management Plans.

[B4] IEEE Std 829-1998, IEEE Standard for Software Test Documentation.

[B5] IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.

[B6] IEEE Std 982.1-1988, IEEE Standard Dictionary of Measures to Produce Reliable Software.

[B7] IEEE Std 982.2-1988, IEEE Guide for the Use of IEEE Standard Dictionary of Measures to Produce Reliable Software.

[B8] IEEE Std 1008-1987 (Reaff 1993), IEEE Standard for Software Unit Testing.

[B9] IEEE Std 1012-1998, IEEE Standard for Software Verification and Validation Plans.

[B10] IEEE Std 1012a-1998, Supplement to IEEE Standard for Software Verification and Validation: Content Map to IEEE/EIA 12207.1-1997.

[B11] IEEE Std 1016-1998, IEEE Recommended Practice for Software Design Descriptions.

[B12] IEEE Std 1028-1997, IEEE Standard for Software Reviews.

[B13] IEEE Std 1058-1998, IEEE Standard for Software Project Management Plans.

[B14] IEEE Std 1063-1987 (Reaff 1993), IEEE Standard for Software User Documentation.

[B15] IEEE Std 1219-1998, IEEE Standard for Software Maintenance.

[B16] IEEE Std 1228-1994™, IEEE Standard for Software Safety Plans.

[B17] IEEE/EIA 12207.0-1996, Guide for Information Technology—Software Life Cycle Processes.

[B18] IEEE/EIA12207.1-1997, Guide for Information Technology—Software Life Cycle Processes—Life Cycle Data.

---

[2]The IEEE standards or products referred to in Annex A are trademarks owned by the Institute of Electrical and Electronics Engineers, Incorporated.