

IEEE Standard Classification for Software Anomalies

Sponsor

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

Reaffirmed September 11, 2002
Approved December 2, 1993

IEEE Standards Board

Abstract: A uniform approach to the classification of anomalies found in software and its documentation is provided. The processing of anomalies discovered during any software life cycle phase are described, and comprehensive lists of software anomaly classifications and related data items that are helpful to identify and track anomalies are provided. This standard is not intended to define procedural or format requirements for using the classification scheme. It does identify some classification measures and does not attempt to define all the data supporting the analysis of an anomaly.

Keywords: anomaly, category, classification, classification process, supporting data item

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

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

Print: ISBN 1-55937-708-9, SH94399
PDF: ISBN 0-7381-0406-X, SS94399

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the Technical Committees of the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason IEEE and the members of its technical committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

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

Secretary, IEEE Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

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

Introduction

(This introduction is not a part of IEEE Std 1044-1993, IEEE Standard Classification for Software Anomalies.)

This standard provides a uniform approach to the classification of anomalies found in software and its documentation. It describes the processing of anomalies discovered during any software life cycle phase, and it provides comprehensive lists of software anomaly classifications and related data items that are helpful to identify and track anomalies. The minimum set of classifications deemed necessary for a complete data set are indicated as mandatory. More detailed classifications are provided for those projects that require more rigor. These lower levels of detail are shown as optional. Application of less than the mandatory set of classifications is not recommended as this may result in insufficient detail for meaningful data collection and analysis.

Some guidelines on how to apply the standard are provided in 4.1. In addition, annex A provides an example anomaly reporting mechanism that describes how to apply the classification scheme. The classification scheme in this standard considers the environment and activity in which the anomaly occurred, the symptoms of the anomaly, the software or system cause of the anomaly, whether the anomaly is a problem or an enhancement request, where the anomaly originated (by phase and document), the resolution and disposition of the anomaly, the impact of several aspects of the anomaly, and the appropriate corrective action.

Collecting the data described in this standard provides valuable information that has many useful applications. Software is usually the most expensive item in computer systems. It is also well documented that the earlier within the software life cycle a problem is discovered, the cheaper it is to fix. This encourages the use of tools, techniques, and methodologies to find problems sooner. Standard anomaly data is necessary to evaluate how well these tools, techniques, and methodologies work. This data can also identify when in a project's life cycle most problems are introduced. Distinctions between enhancements and problems in the software help make the decisions as to which anomalies are addressed first, category of funding, etc. Anomaly data can also assist in the evaluation of reliability and productivity measures.

At the time this standard was completed, the Classification Standards Working Group had the following membership:

Richard Evans, *Chair* **Cynthia Brehmer, *Co-chair***
Jaya Carl, *Secretary*

John B. Bowen
Lynn K. Broderson
Linda Clemens
Richard J. Gale

Myron Lipow
Mary Mikhail
Patricia Pratt
David Schucker

David Simkins
Vijaya Srivastava
Theodore Sullivan
Greg Ward

The following persons were on the balloting committee:

A. Frank Ackerman	Anne Geraci	Geraldine Neidhart
Eleanor Antreassian	Jean A. Gilmore	Dennis Nickle
Bruce M. Bakken	Shirley A. Gloss-Soler	Michael T. Perkins
Jack N. Barnard	Paul Grizenko	William E. Perry
Boris Beizer	L. M. Gunther	Ron Pfaff
H. R. Berlack	Virl Haas	Donald J. Pfeiffer
Barry Boehm	Peter J. Harvey	John N. Postak
Kathleen L. Briggs	John Horch	Robert M. Poston
Christian Brunelle	Laurel V. Kaleda	Jock Rader
Fletcher J. Buckley	Adi Kasad	Arthur S. Robinson
Elliot J. Chikofsky	R. A. Kessler	Frances A. Ruhlman
Jung K. Chung	Tom Kurihara	Margaret Rumley
Francois Coallier	Lak-Ming Lam	Norman Schneidewind
Stewart Crawford	John B. Lane	Leonard W. Seagren
Patricia W. Daggett	Robert A. Lane	Harlan K. Seyfer
James Dobbins	F. C. Lim	Anthony F. J. Sgarlatti
David Doty	Ben Livson	Robert W. Shillato
William P. Dupras	Donald Lundry	Jacob Slonim
Robert E. Dwyer	Austin J. Maher	Wayne Smith
Kenneth Dymond	Kartik C. Majumdar	Terrence L. Tillmanns
Caroline L. Evans	David M. Marks	David B. Turner
James R. Evans	Philip C. Marriott	William Stephen Turner
John W. Fendrich	Darrell Marsh	Ralph Wachter
Glenn S. Fields	Glen A. Meldrum	Dolores R. Wallace
Violet Foldes	Edward F. Miller	John P. Walter
Kenneth A. Foster	Celia H. Modell	Andrew H. Weigel
Richard Fries	Gary D. Moorhead	Bill Wong
David Gelperin	Gene T. Morun	Dennis L. Wood
	Robert C. Natale	

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

Wallace S. Read, *Chair*

Donald C. Loughry, *Vice Chair*

Andrew G. Salem, *Secretary*

Gilles A. Baril	Jim Isaak	Don T. Michael*
José A. Berrios de la Paz	Ben C. Johnson	Marco W. Migliaro
Clyde R. Camp	Walter J. Karplus	L. John Rankine
Donald C. Fleckenstein	Lorraine C. Kevra	Arthur K. Reilly
Jay Forster*	E. G. "Al" Kiener	Ronald H. Reimer
David F. Franklin	Ivor N. Knight	Gary S. Robinson
Ramiro Garcia	Joseph L. Koepfinger*	Leonard L. Tripp
Donald N. Heirman	D. N. "Jim" Logothetis	Donald W. Zipse

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

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

Rachel A. Meisel
IEEE Standards Project Editor

Contents

CLAUSE	PAGE
1. Overview.....	1
1.1 Background.....	1
1.2 Scope.....	2
2. References.....	2
3. Definitions.....	3
4. Classification standard.....	4
4.1 Classification process.....	4
4.2 Standard classification scheme	6
Annex A Example anomaly report	24

IEEE Standard Classification for Software Anomalies

1. Overview

1.1 Background

This standard is based on several definitions, concepts, and processes that need to be understood prior to its use. These are discussed in the following paragraphs. Formal definitions can be found in clause 3.

A key term in this standard is anomaly. An anomaly is any condition that departs from the expected. This expectation can come from documentation (requirements specifications, design documents, user documents, standards, etc.) or from someone's perceptions or experiences. An anomaly is not necessarily a problem in the software product; it may be manifesting correct behavior in which case changing the software would be an enhancement. An anomaly may also be caused by something other than the software. For reasons of semantics, use of the word *anomaly* is preferred over the words *error*, *fault*, *failure*, *incident*, *flaw*, *problem*, *gripe*, *glitch*, *defect*, or *bug* throughout this standard because it conveys a more neutral connotation.

The methodology of this standard is based on a process (sequence of steps) that pursues a logical progression from the initial recognition of an anomaly to its final disposition. Each step interrelates with and supports the other steps. This process is graphically displayed in figure 1, and a complete discussion of the classification process is in 4.1. This standard does not attempt to enforce any particular anomaly processing procedure other than to identify the basic processes an anomaly goes through. It is expected that users will modify the process based on their organization's procedures. Subclause 4.2 describes the classification scheme and supporting data items useful for identifying, tracking, and resolving anomalies.

1.2 Scope

This standard is applicable to any software, including critical computer software, commercial applications, system software, support software, testware, and firmware during any phase of a system's life cycle. This standard defines the minimum requirements for classifying anomalies, as well as providing additional classifications for projects requiring greater detail. The mandatory classifications are the minimum requirements necessary to establish a standard terminology for anomalies within or between projects and organizations.

To accomplish the classification task, this standard documents the following subjects:

- a) Definitions of terms not provided in IEEE Std 610.12-1990¹
- b) A basic process (sequence of steps) for classifying and establishing categories of anomalies relating to software products and providing related data and information
- c) A standard set of categories and classifications, either mandatory or optional
- d) A list of supporting data items

This standard identifies those essential (mandatory) categories needed to establish a common definition. The categories provide a common terminology and concepts to communicate among projects, software development environments, and personnel. It is assumed that all applicable classifications within a mandatory category shall be used for compliance to this standard. This standard also provides categories for additional detail that are not necessarily essential to all projects. These additional categories are identified as optional. This standard is not intended to define procedural or format requirements for using the classification scheme. This standard identifies some classification measures and does not attempt to define all the data supporting the analysis of an anomaly.

The classifications are layered with greater detail in each subsequent layer. For instance, in table 1b the *Project phase* category has two levels of classifications, and in table 3c the *Type* category has three levels of classifications. Each project using this classification scheme may choose the level of classification appropriate to that project. Only the first level of classifications is mandatory for each mandatory category.

The classifications in this standard are not exhaustive. Some projects will have requirements that are unique. It is assumed that in those situations the project members will create project-specific categories and/or classifications in addition to the classifications stated in this standard. Anomaly reports generated using the classifications stated in this standard facilitate the generation of statistics and information for trend analysis.

2. References

This standard shall be used with the following publications:

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

Brehmer, C. L., Carl, J. R., "Incorporating IEEE Std 1044 into Your Anomaly Tracking Process," *Crosstalk*, no. 40, pp. 9–16, Jan. 1993.

¹Information on references can be found in clause 2.

3. Definitions

Definitions of terms used in this standard are consistent with IEEE Std 610.12-1990, with the few exceptions noted below. Note that the term *anomaly* as defined in this standard expands upon the limited definition included in IEEE Std 610.12-1990. This standard's definition of anomaly includes deviations from the user's perceptions or experiences.

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 category: An attribute of an anomaly to which a group of classifications belongs.

3.3 classification: A choice within a category.

3.4 classification process: The classification process is a series of activities, starting with the recognition of an anomaly through to its closure. The process is divided into four sequential steps interspersed with three administrative activities. The sequential steps are as follows:

- a) Step 1: Recognition
- b) Step 2: Investigation
- c) Step 3: Action
- d) Step 4: Disposition

The three administrative activities applied to each sequential step are as follows:

- a) Recording
- b) Classifying
- c) Identifying impact

3.5 mandatory category: A category that is essential to establish a common definition and to provide common terminology and concepts for communication among projects, business environments, and personnel.

3.6 optional category: A category that provides additional details that are not essential but may be useful in particular situations.

3.7 supporting data item: Data used to describe an anomaly and the environment in which it was encountered.

4. Classification standard

This standard refers to the anomaly classification process shown in figure 1. The standard classification scheme is described in 4.2. The standard categories and classifications are shown in the odd-numbered tables. The supporting data items are shown in the even-numbered tables.

Because the classification scheme depends on an understanding of the basic classification process, the classification process will be discussed first.

4.1 Classification process

The classification process is a series of activities, starting with the recognition of an anomaly through to its closure. The process is divided into four sequential steps interspersed with three administrative activities. The sequential steps are as follows:

- a) Step 1: Recognition
- b) Step 2: Investigation
- c) Step 3: Action
- d) Step 4: Disposition

The three administrative activities applied to each sequential step are as follows:

- a) Recording
- b) Classifying
- c) Identifying impact

Each of the steps and administrative activities is discussed below. Figure 1 illustrates the process and interrelationships between the activities. It is anticipated that in the processing of any one anomaly these steps may be repeated several times. It is also expected that individual organizations implementing this standard will use procedures that accomplish the standard's objective but may call for additional detail, a different sequence of steps, and/or may specify organizational involvement.

4.1.1 Recognition

The recognition step occurs when an anomaly is found. Recognition of an anomaly may be made by anyone involved in the development, evaluation, use, support, or maintenance of software, regardless of where in the software life cycle the anomaly is discovered.

- a) *Recording the recognition.* When an anomaly is recognized, supporting data items are recorded to identify the anomaly and the environment in which it occurred. The applicable supporting data items indicated in table 2 are recorded as part of the recognition process.
- b) *Classifying the recognition.* Certain important attributes of an anomaly are observed and shall be identified during the recognition process. Tables 1a–1f contain the categories of information and the scheme by which these attributes shall be classified.
- c) *Identifying impact.* The person discovering the anomaly shall identify his or her perception of the impact of the anomaly. Tables 7a–7i list the categories used for impact identification.