

Recommendations for

Achievement of quality in software

Committees responsible for this British Standard

The preparation of this British Standard was entrusted by the information Systems Technology Standards Policy Committee (IST/-) to Technical Committee IST/15, upon which the following bodies were represented:

- British Computer Society
- Business Equipment and Information Technology Association
- Computing Services Association
- Department of Trade and Industry
- Electronic Engineering Association
- HM Treasury (Central Computer and Telecommunications Agency)
- Institute of Scientific and Technical Communicators
- Institution of Electrical Engineers
- Ministry of Defence
- National Computing Centre Ltd.
- User Standards Forum for Information Technology (Institute of Data Processing Management)

This British Standard, having been prepared under the direction of the Information Systems Technology Standards policy Committee, was published under the authority of the Board of BSI and comes into effect on 28 February 1991

© BSI 01-1999

The following BSI references relate to the work on this standard:
Committee reference IST/15
Draft for comment 86/64000 DC

ISBN 0 580 18843 4

Amendments issued since publication

Amd. No.	Date	Comments

Contents

	Page
Committees responsible	Inside front cover
Foreword	iii
1 Scope	1
2 Definitions	1
3 Pre-development activities	2
3.1 The customer's specification of requirements	2
3.2 The supplier's response	2
3.3 The procurement specification	2
3.4 Software life cycle management planning	2
3.5 Planning for software quality	3
4 Suppliers' codes of practice	3
4.1 General	3
4.2 Project management	3
4.3 Design techniques and methods	3
4.4 Programming techniques and methods	3
4.5 Software development facilities	3
4.6 Documentation	4
4.7 Configuration management	4
4.8 Design review	4
4.9 Tests and trials	5
4.10 Transfer to customer	5
4.11 Post-design services/maintenance	5
4.12 Sub-contracting of software	5
5 Software quality procedures	6
5.1 General	6
5.2 Project management	6
5.3 Design techniques and methods	6
5.4 Programming techniques and methods	6
5.5 Software development facilities	6
5.6 Documentation	7
5.7 Configuration management	7
5.8 Design review	8
5.9 Tests and trials	8
5.10 Transfer to customer	9
5.11 Post-design services/maintenance	9
5.12 Sub-contracting of software	9
Appendix A Pre-contractual activities	10
Appendix B Planning for software quality	11
Appendix C Design and programming techniques and methods	12
Appendix D Documentation	13
Appendix E Configuration management	15
Appendix F Design reviews	19
Appendix G Tests	21
Appendix H Trials procedure	22
Appendix J Transfer to customer	22
Appendix K Sub-contracting of software	23

	Page
Figure 1 — Typical fault reporting procedure	16
Figure 2 — Typical software modification procedure	17
Figure 3 — Typical document modification procedure	18
Publications referred to	Inside back cover

Foreword

This British Standard was prepared under the direction of the Information Systems Technology Standards Policy Committee. Through cooperation with the Ministry of Defence this British Standard is based on, and is technically equivalent to DEF STAN 00-16 Issue 1, February 1984: Guide to Achievement of Quality in Software¹⁾.

The DEF Standard has been prepared primarily for operational defence application and most of the differences introduced have the purpose of expanding the scope so that this British Standard is applicable to the quality of software in a wide range of manufacturing and service industries.

This British Standard addresses the procurement cycle and covers subjects which are important to software quality. It is assumed that the more general aspects of quality management required by BS 5750 are established practice.

For the achievement of quality in software it is necessary to satisfy three principal aims:

- a) a clear definition of what is to be achieved and when (specifications and plans);
- b) a description of the activities and functions that need to be performed (codes of practice);
- c) the control and monitoring of the way in which activities/functions are performed (quality control).

Collectively these aims can be summed up in the term “quality”, as defined in BS 4778.

This standard provides guidance as to how these aims may be achieved to help those responsible for software based projects to meet their commitments effectively, efficiently and economically. The degree and formality of implementation, and the extent of the quality activities, will vary according to the needs of the project. The methods used to implement the principles identified in this standard will also vary according to the size and complexity of the task, and the structure and management organization of the particular organization.

Care has been taken in this standard to identify quality activities without pre-judging the question of who should carry them out. Reference to organizational structures such as “quality departments” have been avoided so that there is flexibility regarding the organizational methods used to satisfy the principles expressed in this document. The quality functions can be integrated with the project tasks provided adequate functional independence is maintained.

As with any engineering activity it is important that suitable codes of practice are specified so that uniform approaches and methods can be applied by designers and associated staff, thus enabling both an assessment of achievement and of quality to be made. Many of the fundamental activities that have a direct bearing on the quality of the final software will be defined by the codes of practice. One of the significant quality functions is to ensure that activities and standards set out in the codes of practice are being followed and that the requirements of the contract, in terms of quality, are defined and satisfied. The extent and nature of these activities should be defined in appropriate quality procedures.

When this standard was approved for publication, work was proceeding in a number of organizations to prepare related informative documents. For example, additional information is published in:

- a) STARTS Purchasers’ Handbook²⁾.

¹⁾ Available from the head of section, STAN-8, Room 5150A, Kentigern House, 65 Brown Street, Glasgow G2 8EX.

²⁾ Available from NCC Publications, National Computing Centre, 61 Oxford Road, Manchester M1 7ED.

b) Draft Interim Defence Standard 00-55 (May 1989) Requirements for the Procurement of Safety Critical Software in Defence Equipment³⁾.

In addition, the International Organization for Standardization (ISO) is preparing a standard covering the application of BS 5750-1 to software.

A British Standard does not purport to include all the necessary provisions of a contract. Users of British Standards are responsible for their correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, pages i to iv, pages 1 to 24, an inside back cover and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

³⁾ Available from the head of section, STAN-8, Room 5150A, Kentigern House, 65 Brown Street, Glasgow G2 8EX.

1 Scope

This British Standard gives recommendations:

- a) for a software quality control system intended to meet the requirements of BS 5750; and
- b) on the practical realization of these requirements to ensure that software is of the required quality.

The principles expressed in this standard are applicable to all types of procurement involving software, ranging from feasibility study through full development to in-service use and maintenance support.

The recommendations in this standard are applicable to all types of software (including firmware).

Many recommendations of this standard are also applicable to the maintenance of software by the user during its in-service life.

This standard addresses pre-contract activities because they have a direct effect upon the quality of software.

This standard does not cover the ramifications of safety critical software, where extra or more intensive quality practices need to be established to meet the approval of the relevant safety assessment authority.

NOTE 1 The words "supplier" and "customer" can be used to mean either separate organizations or separate groups within organizations.

NOTE 2 Throughout this standard it is recommended that various data are kept. A recommended method of doing this is given in BS 5760-1, with a view to building up a data library. A further Part of BS 5760 is in course of preparation that will give guidance on how to use these data to improve the next and subsequent products.

NOTE 3 Appendices A to K give detailed advice supporting the actions outlined in clauses 5 to 7.

NOTE 4 The titles of the publications referred to in this standard are listed on the inside back cover.

2 Definitions

For the purposes of this British Standard the following definitions apply.

2.1

walk through

a review process in which a designer or programmer leads one or more other members of the development team through a segment of design or code that he or she has written, while the other members ask questions and make comments about technique, style, possible errors, violation of development standards and other problems

NOTE This definition is identical with that in IEEE Standard 729 Glossary of software engineering terminology⁴⁾.

2.2

software

software covers all instructions and data which are input to a computer to cause it to function in any mode; it includes operating systems, supervisory systems, compilers and test routines, as well as application programs. Software includes the documents used to define and describe the program (including flow charts, network diagrams and program listings) and also covers specifications, test plans, test data, test results and user instructions

2.3

firmware

software code or logic implanted in non-volatile hardware device
(i.e. firmware = device + software/logic)

2.4

maintenance

any activity intended to eliminate faults or keep hardware or programs in satisfactory working condition, including tests, measurements, adjustments and repairs

2.5

quality

the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs

NOTE This definition is identical with that in BS 4778-1.

2.6

quality control

the operational techniques and activities which are used to fulfil requirements for quality

NOTE This definition is identical with that in BS 4778-1.

2.7

quality assurance

all those planned and systematic actions necessary to provide adequate confidence that a product or service will satisfy given requirements for quality

NOTE This definition is identical with that in BS 4778-1.

2.8

customer

the agency placing the requirement with the supplier

2.9

user

the individual or group ultimately making use of the output associated with the subject software

⁴⁾ Available from BSI Sales Department, Linford Wood, Milton Keynes MK14 6LE.

**2.10
supplier**

the agency tasked with generating software for the customer

**2.11
procurement**

the act of procuring material, i.e. software, for the customer

**2.12
procurement specification**

the specification provided by the customer reflecting the customer's requirements for the software

**2.13
post development**

subsequent development undertaken on completion of the initial development

**2.14
fault**

an accidental condition that causes the software to fail to perform its required function

NOTE The terms project management; project manager; and quality management in the context of this document refer to the supplier's organization.

3 Pre-development activities**3.1 The customer's specification of requirements**

It is essential to establish and document system requirements in a form that is understood by both the customer and prospective suppliers (see BS 5515). The first step is to produce a specific statement of requirements to which potential suppliers can respond. This statement should specify for example, the function of the system, the performance required and all environmental and design constraints including those applicable to software (see appendix A). BS 6719 gives more detailed guidance.

3.2 The supplier's response

The supplier should be required to respond with proposals that describe, point by point and in the order given in the specification of requirements, how they would meet these requirements. The supplier's proposals should also describe how they would plan and manage the project and assure the quality of the system (see appendix A). A procurement specification can then be produced.

3.3 The procurement specification

Procurement specifications may be produced by the customer or by a supplier but should, in any case, be subject to mutual agreement and should define what is to be supplied and how it is to be accepted. It is important that aims or objectives should be clearly differentiated from firm requirements. Some trade-off aspects may be left to be determined during the course of the development, in which case the requirement should be defined in overall terms of system performance.

There are a number of disciplines and methodologies used in the production of procurement specifications, all of which are aimed at securing standardization and precision in the definition of the requirement. Changes to procurement specifications should be the subject of formal control procedures. These should be linked with similar controls of defining specifications and acceptance testing specifications that will be raised in the course of implementation of the task. Only in this way can the implications of changes be properly examined and established.

NOTE Particular care is needed in the definition of the interfaces between hardware and software and with the terminology used in order to avoid misunderstandings between the disciplines.

3.4 Software life cycle management planning

It is essential that the user's own long term plans for post-development software are made known to the supplier at the earliest opportunity. This may be done through a software life cycle management plan the purpose of which is to define the total system and to identify major events (or milestones) applying to software that will occur in the system life both during and after development together with their timescales. For each milestone identified, the plan should specify the resource required and the agency responsible. The production of a successful software life cycle management plan depends on close coordination between the user, the customer and the supplier.

3.5 Planning for software quality

Whenever a software requirement exists or arises it is essential that a policy agreed by the design authority for software quality is produced as a basis upon which the operation can be planned, both technically and financially. The technical aspects should include planning the most effective use of manpower and technical facilities available (e.g. computer resources, software tools). The financial aspects should involve planning the most effective use of the resources allocated to software quality, and the development of a strategy so that only essential work is carried out. The work should achieve planned quality targets or milestones. Wherever possible, the technical facilities available should be used to assist the quality function in order to release the available manpower for other work (such as design reviews and audits). A practical method of combining all these aspects can be to develop a quality plan for each project (see appendix B).

4 Suppliers' codes of practice

4.1 General

Any quality control arrangements should be based on the documented methods to be employed by the design and development teams as defined by the supplier. Codes of practice provide guidance on good practices and should be regularly reviewed in the light of experience. Organizations undertaking a large number of software-related projects are likely to prepare an overall document outlining the general policies to be implemented on all projects. Individual project codes of practices would then be produced as required detailing many differences or additions to that policy. The main activities which should be documented by the supplier are given in 4.2 to 4.12.

4.2 Project management

4.2.1 Methods

The methods by which management reviews the development requirements should establish a strategy to meet these requirements including identification of milestones. Methods should also be defined for the breakdown of work into a hierarchical structure, allocating and scheduling resources, determining applicable codes of practice and identification of sub-contract activities.

4.2.2 Project planning

Planning information should include a project timeplan which identifies events, milestones and resources. The agreed codes of practice should define methods of plan presentation and dissemination, review, update, and the assignment of work.

4.2.3 Project reporting

Procedures for monitoring, reporting and progressing against the plan, including the identification and control of deviations, should be detailed.

4.3 Design techniques and methods

The code of practice should identify and document all of the acceptable design techniques and methods that are available for use (see appendix C).

NOTE Project plans should specify those design techniques and methods which are to be used.

4.4 Programming techniques and methods

Programming techniques and methods to be used should be defined and documented, e.g. in a programmer's manual. As well as describing approved programming practices, the document should list any programming practices which are prohibited and explain why. A section on languages should either include or refer to a full definition of both the language and the compiler in use. Other sections should include programming and coding conventions, standard algorithms and/or values to be used, optimization methods and debugging aids and their use (see appendix C).

4.5 Software development facilities

Procedures should be defined for the control, monitoring and review of software development facilities and their operation and maintenance. The facilities may, for example, include a computer bureau, the system hardware, a separate dedicated system and software tools. The following items should be considered for inclusion in the supplier's code of practice:

- a) software development facility operating procedures;
- b) configuration management of the support software;
- c) user information notices;
- d) software development facility maintenance;
- e) confidence checking of the support software;
- f) data protection;
- g) security aspects;
- h) media identification and control;
- i) environmental and storage recommendations.

4.6 Documentation

Documentation is the principal means by which software design and functional information is disseminated and can be usefully divided into four categories (see also appendix D) as follows.

- a) *Design documentation.* A description of the documentation used during the design process for defining the levels of software design created to meet the operational requirement together with any associated design requirements. The documentation should also contain a record of the achievements of the design process.
- b) *Test documentation.* A description of the documentation used during the test, integration and acceptance phases of software development to define the test requirements which the software should meet as it is progressively integrated into a complete package. This documentation should include test records.
- c) *Operational documentation.* A description of the form and content of the documentation required by the user or operator of the system.
- d) *Maintenance documentation.* A description of the form and content of the documentation required by the software maintainer to enable effective and economical maintenance.

NOTE Design and test documentation should be closely linked with the design techniques and methods used.

4.7 Configuration management

4.7.1 General

Configuration management of the software element of a system should be achieved to the same degree as that for the hardware and, wherever possible, using the same methods; the control of the two elements should be fully integrated. One area where integration of control is particularly important is in the use of “firmware”. Procedures should be selected to ensure that control of the software is exercised, as appropriate to the type of system and the stage of development. The application of these controls is known as software configuration control. (Further guidance is given in BS 6488.) Particular attention should be paid in the supplier’s code of practice to defining the point at which documentation and programs are to be brought under configuration control. There are four main topics to be documented in the supplier’s code of practice (see also appendix E).

4.7.2 Fault reporting

Faults which arise within the system should be the subject of procedures which ensure that not only are the faults reported, but that timely and effective corrective action is taken. The action taken, and authorization, should also be recorded.

4.7.3 Software modification

Any modifications to the software should be the subject of a procedure that records the history of the modification, the action taken, re-test requirements, and the authority to implement the modification. Completion of the action should also be recorded.

4.7.4 Documentation modification

Software documentation is an integral part of the software and should be included within the reporting and change procedures. However, it is important that software changes are validated as correct before changes are made to issued documents.

4.7.5 Library

NOTE Within the context of this standard, the function of a library is to act as an organized repository and control mechanism for software components.

The library function should cover all aspects of software configuration control with emphasis on:

- a) acceptance of software into the repository;
- b) controlled release of software from the repository;
- c) retention of master copies of software;
- d) identification of software;
- e) physical security of software;
- f) linked identification and simultaneous revision of interdependent elements of software;
- g) ensuring that changes to one version are carried forward to other versions of software where applicable;
- h) maintaining a historical record of software evolution;
- i) maintenance of a recall/disposal procedure for out-of-issue programs.

Library functions and their operating procedures should be documented in the supplier’s code of practice.

4.8 Design review

The supplier’s code of practice should cover two basic principles.

- a) There should be a series of documented reviews of design work corresponding to the stages in the development life cycle, and progress to the next stage should be dependent on the work under review being approved.
- b) The design work should be reviewed by someone who did not originate the design, which may, in some circumstances, include the customer.