NORME INTERNATIONALE

INTERNATIONAL STANDARD

CEI
IEC
60300-3-6

Première édition
First edition
1997-11

Gestion de la sûreté de fonctionnement –

Partie 3:
Guide d'application –
Section 6: Aspects logiciels de la sûreté de fonctionnement

Dependability management –

Part 3:
Application guide –
Section 6: Software aspects of dependability

Numéro de référence
Reference number
CEI/IEC 60300-3-6: 1997

Numéros des publications

Depuis le 1er janvier 1997, les publications de la CEI sont numérotées à partir de 60000.

Publications consolidées

Les versions consolidées de certaines publications de la CEI incorporant les amendements sont disponibles. Par exemple, les numéros d'édition 1.0, 1.1 et 1.2 indiquent respectivement la publication de base, la publication de base incorporant l'amendement 1, et la publication de base incorporant les amendements 1 et 2.

Validité de la présente publication

Le contenu technique des publications de la CEI est constamment revu par la CEI afin qu'il reflète l'état actuel de la technique.

Des renseignements relatifs à la date de reconfirmation de la publication sont disponibles dans le Catalogue de la CEI.

Les renseignements relatifs à ces révisions, à l'établissement des éditions révisées et aux amendements peuvent être obtenus auprès des Comités nationaux de la CEI et dans les documents ci-dessous:

- **Bulletin de la CEI**
- **Annuaire de la CEI**
  Accès en ligne*
- **Catalogue des publications de la CEI**
  Publié annuellement et mis à jour régulièrement
  (Accès en ligne)*

Terminologie, symboles graphiques et littéraux

En ce qui concerne la terminologie générale, le lecteur se reportera à la CEI 60050: *Vocabulaire Électrotechnique International* (VEI).

Pour les symboles graphiques, les symboles littéraux et les signes d'usage général approuvés par la CEI, le lecteur consultera la CEI 60027: *Symboles littéraux à utiliser en électrotechnique,* la CEI 60417: *Symboles graphiques utilisables sur le matériel. Index, relevé et compilation des feuilles individuelles,* et la CEI 60617: *Symboles graphiques pour schémas.*

Publications de la CEI établies par le même comité d'études

L'attention du lecteur est attirée sur les listes figurant à la fin de cette publication, qui énumèrent les publications de la CEI préparées par le comité d'études qui a établi la présente publication.

* Voir adresse «site web» sur la page de titre.

Numbering

As from the 1st January 1997 all IEC publications are issued with a designation in the 60000 series.

Consolidated publications

Consolidated versions of some IEC publications including amendments are available. For example, edition numbers 1.0, 1.1 and 1.2 refer, respectively, to the base publication, the base publication incorporating amendment 1 and the base publication incorporating amendments 1 and 2.

Validity of this publication

The technical content of IEC publications is kept under constant review by the IEC, thus ensuring that the content reflects current technology.

Information relating to the date of the reconfirmation of the publication is available in the IEC catalogue.

Information on the revision work, the issue of revised editions and amendments may be obtained from IEC National Committees and from the following IEC sources:

- **IEC Bulletin**
- **IEC Yearbook**
  On-line access*
- **Catalogue of IEC publications**
  Published yearly with regular updates
  (On-line access)*

Terminology, graphical and letter symbols

For general terminology, readers are referred to IEC 60050: *International Electrotechnical Vocabulary* (IEV).

For graphical symbols, and letter symbols and signs approved by the IEC for general use, readers are referred to publications IEC 60027: *Letter symbols to be used in electrical technology*, IEC 60417: *Graphical symbols for use on equipment. Index, survey and compilation of the single sheets* and IEC 60617: *Graphical symbols for diagrams.*

IEC publications prepared by the same technical committee

The attention of readers is drawn to the end pages of this publication which list the IEC publications issued by the technical committee which has prepared the present publication.

* See web site address on title page.

NORME INTERNATIONALE

INTERNATIONAL STANDARD

CEI
IEC
60300-3-6

Première édition
First edition
1997-11

Gestion de la sûreté de fonctionnement –

Partie 3:
Guide d'application –
Section 6: Aspects logiciels de la sûreté de fonctionnement

Dependability management –

Part 3:
Application guide –
Section 6: Software aspects of dependability

International Electrotechnical Commission 3, rue de Varembé Geneva, Switzerland
Telefax: +41 22 919 0300 e-mail: inmail@iec.ch IEC web site http: //www.iec.ch

Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

CODE PRIX
PRICE CODE U

Pour prix, voir catalogue en vigueur
For price, see current catalogue

SOMMAIRE

CONTENTS

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

_____

GESTION DE LA SÛRETÉ DE FONCTIONNEMENT –

Partie 3: Guide d'application – Section 6: Aspects logiciels de la sûreté de fonctionnement

AVANT-PROPOS

1) La CEI (Commission Electrotechnique Internationale) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI, entre autres activités, publie des Normes internationales. Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible un accord international sur les sujets étudiés, étant donné que les Comités nationaux intéressés sont représentés dans chaque comité d'études.
3) Les documents produits se présentent sous la forme de recommandations internationales. Ils sont publiés comme normes, rapports techniques ou guides et agréés comme tels par les Comités nationaux.
4) Dans le but d'encourager l'unification internationale, les Comités nationaux de la CEI s'engagent à appliquer de façon transparente, dans toute la mesure possible, les Normes internationales de la CEI dans leurs normes nationales et régionales. Toute divergence entre la norme de la CEI et la norme nationale ou régionale correspondante doit être indiquée en termes clairs dans cette dernière.
5) La CEI n'a fixé aucune procédure concernant le marquage comme indication d'approbation et sa responsabilité n'est pas engagée quand un matériel est déclaré conforme à l'une de ses normes.
6) L'attention est attirée sur le fait que certains des éléments de la présente Norme internationale peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

La Norme internationale CEI 60300-3-6 a été établie par le comité d'études 56 de la CEI: Sûreté de fonctionnement.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
56/583/FDIS	56/600/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Les annexes A, B, C, D, E et F sont données uniquement à titre d'information.

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

DEPENDABILITY MANAGEMENT –

Part 3: Application guide –
Section 6: Software aspects of dependability

FOREWORD

1) The IEC (International Electrotechnical Commission) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of the IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, the IEC publishes International Standards. Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. The IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.

2) The formal decisions or agreements of the IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested National Committees.

3) The documents produced have the form of recommendations for international use and are published in the form of standards, technical reports or guides and they are accepted by the National Committees in that sense.

4) In order to promote international unification, IEC National Committees undertake to apply IEC International Standards transparently to the maximum extent possible in their national and regional standards. Any divergence between the IEC Standard and the corresponding national or regional standard shall be clearly indicated in the latter.

5) The IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with one of its standards.

6) Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. The IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 60300-3-6 has been prepared by IEC technical committee 56: Dependability.

The text of this standard is based on the following documents:

FDIS	Report on voting
56/583/FDIS	56/600/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

Annexes A, B, C, D, E and F are for information only.

INTRODUCTION

La sûreté de fonctionnement est un terme générique utilisé pour traduire la disponibilité d'un système ou d'un produit. Cette disponibilité dépend de la fiabilité, de la maintenabilité et de la logistique de maintenance du produit. Dans de nombreux systèmes et produits, la fiabilité, la maintenabilité et la disponibilité font partie des caractéristiques les plus importantes aux yeux des utilisateurs qui recherchent un fonctionnement rentable. La fiabilité et la maintenabilité sont des caractéristiques inhérentes à la conception d'un produit, alors que la logistique de maintenance est un critère indépendant qui influe sur la qualité du service et qui reflète la capacité d'une organisation de maintenance à fournir les ressources nécessaires pour atteindre les objectifs de disponibilité.

Pour qu'un programme de sûreté de fonctionnement soit efficace, sa mise en oeuvre doit être adaptée au produit. Il est recommandé d'inclure ce programme dans celui de gestion globale du projet afin d'obtenir une bonne coordination avec le développement, la fabrication, la vérification et l'utilisation du produit. Il est aussi conseillé que les éléments et les tâches du programme de sûreté de fonctionnement soient cohérents avec les autres programmes de support tels que la gestion de la qualité, la gestion de la configuration, la collecte des données etc.

La gestion de la sûreté de fonctionnement inclut la planification du projet, la spécification, l'analyse de la conception, la vérification et la validation, la réalisation, l'évaluation du produit ou du service ainsi que le retour d'expérience. Les systèmes et produits modernes comportent souvent un logiciel comme entité fonctionnelle, utilisé pour atteindre les objectifs de performances opérationnelles. Le logiciel, inclus dans le système ou dans le produit, est assujetti au processus de gestion de la sûreté de fonctionnement. Le présent guide d'application concerne les aspects logiciels liés à cette gestion. Il donne des directives spécifiques permettant de sélectionner les actions appropriées et de les appliquer aux programmes de sûreté de fonctionnement associés aux produits contenant le logiciel ou aux systèmes configurés par le logiciel et comprenant des éléments matériels.

La disponibilité d'un produit peut être affectée par des défaillances du matériel, des défauts de logiciel ou des erreurs humaines. Un mauvais fonctionnement provoquant des temps d'indisponibilité peut être imputable à des anomalies de conception interne ou à des interférences avec l'extérieur, y compris des erreurs de procédure. Des défaillances du produit peuvent, elles aussi, provenir d'erreurs de conception internes liées à des problèmes de matériel ou de logiciel. Les matériels défectueux ou les pièces usées peuvent être identifiés et localisés, réparés ou remplacés, afin de conserver le niveau de fiabilité du produit. A la différence de la plupart des matériels, un logiciel, une fois créé sous la forme de codes et d'instructions, ne s'use pas et ne se détériore pas. C'est la raison pour laquelle certains processus applicables aux logiciels peuvent être différents de ceux utilisés pour la mise en oeuvre du matériel. Le but de ce guide d'application est de lier le cycle de vie du logiciel à celui du produit, dans le cadre de la gestion de la sûreté de fonctionnement.

La gestion de la sûreté de fonctionnement est définie dans la CEI 60300-1. Les éléments et les tâches du programme correspondant sont spécifiés dans la CEI 60300-2. Le présent guide d'application est complémentaire à la CEI 60300-2 en ce qui concerne la mise en oeuvre du programme de sûreté de fonctionnement des systèmes ou produits dotés d'un logiciel. L'accent est mis sur l'application séquentielle des activités relatives au logiciel associées à la mise en oeuvre de la CEI 60300-2, ainsi que cela est présenté à l'annexe A. L'annexe B présente la sélection des éléments du programme de sûreté de fonctionnement associés aux phases du cycle de vie du logiciel.

Des efforts ont été réalisés pour harmoniser le présent guide d'application avec l'ISO/CEI 12207 sur les processus du cycle de vie du logiciel. Une vue générale du processus du cycle de vie du logiciel est fournie à l'annexe C. Des correspondances sont identifiées à l'annexe D pour faciliter l'association du processus de cycle de vie du logiciel avec les éléments de sûreté de fonctionnement appropriés et les phases du cycle de vie du produit.

INTRODUCTION

Dependability is the collective term describing the availability performance of a system or product. The availability performance is influenced by the reliability, maintainability and maintenance support performance factors. In many systems and products, reliability, maintainability, and availability rank amongst the dominant performance characteristics of importance to the users seeking cost-effective operation. Reliability and maintainability are performance characteristics inherent to the product design. Maintenance support is external to the product itself, and will affect the quality of service. Maintenance support performance reflects the ability of the maintenance organization to provide the necessary resources to sustain a level of maintenance support effort to achieve availability performance objectives.

A dependability programme must be tailored to the product for effective application. The dependability programme should form part of the overall project management programme for proper coordination with product development, manufacture, verification and deployment. Dependability programme elements and tasks should be consistent with the other support programmes such as quality management, configuration management, data collection etc.

The dependability management process includes project planning, specification, design analysis, verification and validation, implementation, evaluation, and data feedback of the product or service. Modern systems and products often contain software as a functional entity to achieve operational performance objectives. The software contained in the system or embedded in the product is subject to the dependability management process. This application guide addresses the software aspects of dependability. It provides specific guidance on the selection and application of relevant activities in dependability programmes associated with products containing software, or systems configured by software with hardware elements.

The availability performance of a product can be affected by hardware failures, software faults, or human errors. Product malfunction causing downtimes can be traceable to its internal design anomalies, or due to external interference including procedural errors. Product failures can arise from internal design faults relating to hardware or software problems. Failed hardware and worn-out parts can be identified and isolated, repaired or replaced to maintain the same level of product reliability. Unlike most physical hardware, software, once created in the form of codes or instructions, will not wear-out or deteriorate. Hence, some of the software processes may be different from those applicable for the hardware implementation. The intent of this application guide is to relate the software life cycle processes with the product life cycle phases within the dependability management framework.

Dependability management is defined in IEC 60300-1. Dependability programme elements and tasks are specified in IEC 60300-2. This application guide complements IEC 60300-2 in terms of dependability programme implementation of systems or products containing software. Emphasis is placed on the time-phase application of relevant software activities associated with the implementation of IEC 60300-2 as shown in annex A. Annex B presents the selection of dependability programme elements associated with the software life cycle phases.

Efforts have been made to harmonize this application guide with ISO/IEC 12207 on software life cycle processes. An overview of the software life cycle processes is provided in annex C. Cross-references are identified in annex D to facilitate association of software life cycle processes with relevant dependability elements and product life cycle phases.

La relation entre la sûreté de fonctionnement (série de normes CEI 60300) et la qualité (série de normes ISO 9000) est traitée dans la CEI 60300-1/ISO 9000-4 et ne sera pas abordée dans le présent guide d'application. Toutefois, il convient de noter, en termes de facteurs de qualité influençant les caractéristiques de sûreté de fonctionnement des éléments de logiciel, les recommandations de l'ISO 9000-3 concernant l'application de l'ISO 9001 aux logiciels. Les correspondances entre la CEI 60300-2 et l'ISO 9000-3 sont présentées à l'annexe E.

Une bibliographie est fournie, à l'annexe F, pour donner des références supplémentaires concernant les aspects logiciels de la sûreté de fonctionnement.

The relationship of dependability (IEC 60300 series of standards) and quality (ISO 9000 series of standards) is addressed in IEC 60300-1/ISO 9000-4 and will not be elaborated in this application guide. However, the guidelines contained in ISO 9000-3 for application of ISO 9001 to software should be noted in terms of quality factors influencing the dependability characteristics of software elements. Cross-references between IEC 60300-2 and ISO 9000-3 are shown in annex E.

A bibliography is provided in annex F for additional references to software aspects of dependability.

GESTION DE LA SÛRETÉ DE FONCTIONNEMENT –

Partie 3: Guide d'application – Section 6: Aspects logiciels de la sûreté de fonctionnement

1 Domaine d'application

Le présent guide d'application est complémentaire à la CEI 60300-2. Il donne des lignes directrices pour la sélection des éléments et des tâches du programme de sûreté de fonctionnement, et leur application aux systèmes et produits comportant un logiciel.

Le présent guide d'application est conçu pour les chefs de projets, les gestionnaires, les concepteurs de produits, les développeurs de logiciels, les spécialistes en sûreté de fonctionnement, les spécialistes de la qualité, les personnels de soutien et les personnels chargés de l'entretien des systèmes, qui contribuent à la sûreté de fonctionnement des produits ou des systèmes.

2 Références normatives

Les documents normatifs suivants contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente section de la CEI 60300-3. Au moment de la publication, les éditions indiquées étaient en vigueur. Tout document normatif est sujet à révision et les parties prenantes aux accords fondés sur la présente section de la CEI 60300-3 sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des documents normatifs indiqués ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur.

CEI 60050(191): 1990, *Vocabulaire électrotechnique international (VEI) – Chapitre 191: Sûreté de fonctionnement et qualité de service*

CEI 60300-1/ISO 9000-4: 1993, *Gestion de la sûreté de fonctionnement – Partie 1: Gestion du programme de sûreté de fonctionnement*

CEI 60300-2: 1995, *Gestion de la sûreté de fonctionnement – Partie 2: Eléments et tâches du programme de sûreté de fonctionnement*

CEI 61160: 1992, *Revue de conception formalisée*
Amendement 1 (1994)

ISO 8402: 1994, *Management de la qualité et assurance de la qualité – Vocabulaire*.

3 Définitions

Pour les besoins de la présente section de la CEI 60300-3, les termes et définitions de la CEI 60050(191) et de l'ISO 8402 s'appliquent.

4 Aspects logiciels

Les aspects logiciels de la sûreté de fonctionnement traitent des considérations spécifiques relatives aux logiciels pour l'établissement et la mise en oeuvre d'un programme de sûreté de fonctionnement, pour un système ou un produit comprenant du logiciel. Le point important est d'obtenir la sûreté de fonctionnement du produit et d'atteindre les objectifs de fiabilité, de maintenabilité et de logistique de maintenance.

DEPENDABILITY MANAGEMENT –

Part 3: Application guide –
Section 6: Software aspects of dependability

1 Scope

This application guide complements IEC 60300-2 and provides guidance for selection and application of dependability elements and tasks with respect to systems or products containing software.

This application guide is intended for use by project managers, contract administrators, product designers, software developers, dependability specialists, quality specialists, support personnel and system maintainers who contribute to the dependability of products or systems.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this section of IEC 60300-3. At the time of publication, the editions indicated were valid. All normative documents are subject to revision, and parties to agreements based on this section of IEC 60300-3 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

IEC 60050(191): 1990, *International Electrotechnical Vocabulary (IEV) – Chapter 191: Dependability and quality of service*

IEC 60300-1/ISO 9000-4: 1993, *Dependability management – Part 1: Dependability programme management*

IEC 60300-2: 1995, *Dependability management – Part 2: Dependability programme elements and tasks*

IEC 61160: 1992, *Formal design review*
Amendment 1 (1994)

ISO 8402: 1994, *Quality management and quality assurance – Vocabulary*

3 Definitions

For the purpose of this section of IEC 60300-3, the terms and definitions of IEC 60050(191) and ISO 8402 apply.

4 Software aspects

The software aspects of dependability deal with the specific software issues in the establishment and implementation of a dependability programme for a system or product containing software. Emphasis is placed on achieving dependability in the product design and performance objectives in reliability, maintainability and maintenance support.

Lorsque l'on veut doter un système ou un produit d'un programme de sûreté de fonctionnement, il est important d'adopter le point de vue du système. Un produit est une entité pouvant inclure des composants matériels et/ou logiciels. Un système est une entité composite pouvant comprendre le produit, le matériel de soutien, le personnel ainsi que les moyens de support et les services associés. Son environnement détermine ses conditions de fonctionnement et les interactions de ses composants. Sa disponibilité est mesurée ou estimée afin de vérifier que les objectifs de sûreté de fonctionnement en termes de fiabilité, maintenabilité et logistique de maintenance sont atteints.

La sûreté de fonctionnement est une mesure globale des caractéristiques d'un système dans son application ou son utilisation réelles, avec ou sans la mise en oeuvre des fonctions logicielles spécifiques qui peuvent faire partie d'un système intégré.

Il convient de noter qu'un logiciel ne peut pas fonctionner de manière isolée, mais qu'il fait partie d'un système et que son rôle est de remplir une application spécifique. C'est un moyen permettant d'atteindre les objectifs de performance d'un système. Un logiciel est caractérisé, en particulier, par sa fonction, son environnement, sa taille, son langage et sa complexité ainsi que par son installation et son processus d'évolution. Les aspects logiciels de la sûreté de fonctionnement ont rapport avec les composants logiciels du système dans le contexte de la performance de la sûreté de fonctionnement du système. Ils ne dépendent pas de la qualité du logiciel en tant que tel. Cette qualité est décrite dans l'ISO/CEI 9126 [1]*.

Les aspects logiciels de la sûreté de fonctionnement dépendent de l'intégrité du composant logiciel pendant le fonctionnement du système. Cette intégrité est inhérente à la conception et associée au confinement du risque. Le risque est une exposition indésirable ou une crainte associée à l'exploitation du système. Le risque est caractérisé par sa probabilité d'occurrence et ses impacts ou conséquences lorsque l'événement survient. La capacité d'un système et de son logiciel à confiner un risque dépend de l'architecture du système, de la tolérance aux pannes et de la rigueur avec laquelle les méthodes appropriées sont appliquées au logiciel. Le niveau d'intégrité sert à caractériser le risque assigné associé au fonctionnement du système à confiner. Lorsqu'un logiciel influe sur les performances d'un système, la relation entre sûreté de fonctionnement et intégrité est étroitement en rapport avec la criticité de l'application logicielle associée aux niveaux d'intégrité assignés.

5 Phases et processus du cycle de vie d'un logiciel

Le cycle de vie d'un logiciel est étroitement lié à celui de son système parent. Une relation typique entre les phases du cycle de vie d'un logiciel et celles d'un produit classique, conforme à la CEI 60300-1/ISO 9000-4, est indiquée à l'annexe A. Un exemple de sélection d'éléments du programme de sûreté de fonctionnement associés aux phases du cycle de vie du logiciel est présenté à l'annexe B.

Le processus du cycle de vie d'un logiciel consiste en un ensemble d'activités ou de tâches planifiées, effectuées pour atteindre le but ou l'objectif déclaré d'un projet. Ce processus implique des activités liées au logiciel, depuis sa conception jusqu'à la fin de son utilisation. Des informations détaillées sont données dans l'ISO/CEI 12207 [2]. L'annexe C donne des notes informatives sur les processus du cycle de vie des logiciels.

L'annexe D montre l'association des processus du cycle de vie des logiciels dans un programme caractéristique de sûreté de fonctionnement. La CEI 60300-2 donne la description des différents éléments d'un programme de sûreté de fonctionnement, en fonction des phases du cycle de vie d'un produit. Le processus du cycle de vie d'un logiciel n'est pas toujours directement lié au cycle de vie du produit. Les relations séquentielles du processus avec les phases ne sont qu'approximatives, car il peut se produire de grandes fluctuations entre les différents projets. C'est ainsi que, par exemple, la version d'un logiciel peut devoir être réalisée avant le début de la fabrication du matériel. Cependant, les différents éléments et tâches de

* Les chiffres entre crochets renvoient à la bibliographie donnée à l'annexe F.

In the application of a dependability programme to a system or product, it is important to address the dependability issue from a system view point. A product is an entity which may contain hardware or software components or both. A system is an integrated composite entity, which may include the product, supply material, personnel, and related support facilities and services. The system environment defines the operating conditions and interactions of the system components. The availability performance of the system is measured or assessed to validate the achievement of stated dependability objectives in terms of reliability, maintainability, and maintenance support.

Dependability is a collective measure of the performances of a system in its actual state of application or use, with or without the operation of specific software functions which may form part of an integrated system.

It should be noted that software cannot function by itself but forms part of a system to provide specific application. Software is a medium for realization of a system performance objective. Software is characterized in particular by its application function, operating environment, size, language and complexity, installation and upgrade processes. The software aspects of dependability address the software components within a system in the context of dependability performance of the system. They do not address the quality of the software as a stand alone item. Software quality is described in ISO/IEC 9126 [1]*.

The software aspects of dependability are associated with the integrity of the software component in system operation. Integrity is an inherent design attribute associated with risk containment. Risk is an undesirable exposure or threat associated with the system operation. Risk is characterized by its probability of occurrence and its impact or consequence of the event outcome. The ability of a system and its software component to contain risk is dependent on the system architecture, fault tolerant design, and the degree of rigour in the application of relevant methods to the software. The integrity level is the assigned risk associated with the system operation which is to be contained. The relationship between dependability and integrity is closely linked by the criticality of software application associated with the assigned integrity levels when dealing with the software affecting system performance.

5 Software life cycle phases and processes

The life cycle of software is very much intertwined with the life cycle of its parent system. A typical relationship of the software life cycle phases and the conventional product life cycle phases in accordance with IEC 60300-1/ISO 9000-4 is shown in annex A. An example for the selection of dependability programme elements associated with the software life cycle phases is presented in annex B.

A software life cycle process is a set of planned activities or tasks performed accordingly to achieve a stated goal or objective of a software project. The process involves activities pertaining to a software product from its conception to the termination of its use. Detailed information is contained in ISO/IEC 12207 [2]. Annex C provides informative notes on the software life cycle processes.

Annex D shows the association of software life cycle processes in a typical dependability programme. IEC 60300-2 describes the various elements of a dependability programme in terms of product life cycle phases. The software life cycle processes do not always relate directly to the product life cycle. The process relationships with respect to time are only approximate as considerable variation may occur between different software projects. For example, in some cases software release has to occur before hardware product manufacturing

* Figures in square brackets refer to the bibliography given in annex F.

sûreté de fonctionnement ont été incorporés pour illustrer la nature des phases dans les programmes correspondants. La représentation à l'aide de phases est utile pour procéder à l'intégration des données du logiciel dans le matériel ou le produit.

Certaines activités de sûreté de fonctionnement et de qualité, liées entre elles, sont associées à la réalisation d'un logiciel ou d'un programme. L'annexe E donne les correspondances entre les éléments et les tâches du programme de sûreté de fonctionnement de la CEI 60300-2 et l'ISO 9000-3 [3].

6 Application des programmes de sûreté de fonctionnement aux produits comprenant un logiciel

6.1 Généralités

Les directives de la CEI 60300-2 se rapportent aux programmes de sûreté de fonctionnement des produits, y compris ceux comportant un logiciel. La présente norme n'est pas une duplication des exigences fondamentales décrites dans la CEI 60300-2. Elle en est le complément et définit les tâches additionnelles requises applicables aux aspects logiciels de la sûreté de fonctionnement. Elle indique comment sélectionner et appliquer les tâches de sûreté de fonctionnement pour formuler et mettre en oeuvre des programmes spécifiques de sûreté de fonctionnement associés aux produits comprenant un logiciel. Les paragraphes suivants correspondent aux éléments et tâches d'un programme de sûreté de fonctionnement spécifique à un projet ou à un produit, décrits dans la CEI 60300-2, afin de faciliter la correspondance entre les deux normes.

6.2 Planification et gestion

6.2.1 Plans de sûreté de fonctionnement

Il convient d'établir les plans de sûreté de fonctionnement conformément à 6.1.1 de la CEI 60300-2.

Il est recommandé que le plan de sûreté de fonctionnement d'un système ou d'un produit comportant un logiciel soit un plan intégré qui indique les tâches correspondantes applicables au système ou au produit, et à ses différents composants matériels et logiciels. Il convient que le plan fasse partie du plan global de gestion du projet.

Il convient que les aspects logiciels de la sûreté de fonctionnement prennent en compte les considérations suivantes:

– les exigences du produit ou du système concernant le logiciel dont ils sont dotés;
– les exigences contractuelles sur les dates prévues de livraison et la planification de déploiement du logiciel concerné;
– la criticité de la fonction du logiciel qui influe sur les caractéristiques du système ou du produit, dans les conditions de fonctionnement réelles;
– la possibilité d'utiliser des logiciels déjà développés ou disponibles en stock;
– le calendrier et les moyens retenus pour le développement du logiciel, si le projet le prescrit;
– l'interface fonctionnelle du logiciel;
– les exigences de documentation;
– les exigences d'essai du logiciel et de son intégration dans le système;
– les exigences de la logistique de maintenance du logiciel;
– les exigences de mise à jour et d'autorisation de livraison du logiciel.

can start. However, the relevant dependability elements and tasks are incorporated here to illustrate the essence of time-phase applications in dependability programmes. This time-phase representation provides a useful framework for integrating software elements into a hardware system or product.

There are related dependability and quality activities associated with the implementation of a software project or programme. Annex E provides cross-referencing of IEC 60300-2 dependability programme elements and tasks to ISO 9000-3 [3].

6 Application of dependability programmes to products containing software

6.1 General

The guidance given in IEC 60300-2 provides a basis for dependability programmes for products including those containing software. This standard does not duplicate the basic requirements described in IEC 60300-2 but identifies additional task requirements relevant to software aspects of dependability, to complement IEC 60300-2. Guidance is provided on the selection and application of dependability tasks for formulating and implementing specific dependability programmes of products containing software. The following subclauses correspond to the project or product specific dependability programme elements and tasks described in IEC 60300-2 to facilitate cross-referencing.

6.2 Planning and management

6.2.1 Dependability plans

Dependability plans should be established in accordance with 6.1.1 of IEC 60300-2.

A dependability plan for a system or product containing software should be an integrated plan that shows the relevant dependability tasks applicable to the system or product, and its constituent hardware and software components. The dependability plan should form part of the overall project management plan.

Planning for software aspects of dependability should take into consideration:

– the system or product requirements pertaining to the software contained in the system or product;
– specific contract requirements affecting the delivery targets and deployment schedule of the software involved;
– the criticality of the software function affecting system or product performance in actual operating environment;
– the feasibility of reusing pre-developed software or off-the-shelf software products;
– timing and resources for the software development if required by the project;
– functional interface of the software involved;
– documentation requirements;
– software test and system integration requirements;
– software maintenance support requirements;
– software update and release requirements.

6.2.2 Gestion des décisions concernant le projet

Il est recommandé que la gestion des décisions concernant le projet soit conforme à 6.1.2 de la CEI 60300-2.

Il est conseillé que la gestion d'un projet de logiciel fasse partie du plan global de gestion du projet concernant le système ou de celui du produit comportant ce logiciel. Il convient que les points-clés prévus pour le logiciel se traduisent par un ensemble coordonné de dates de livraisons cohérentes afin de faciliter la prise de décision concernant la gestion lors de revues planifiées.

6.2.3 Gestion de la traçabilité

Il est recommandé que la gestion de la traçabilité soit conforme à 6.1.3 de la CEI 60300-2.

Il est recommandé que la traçabilité des travaux de développement du logiciel et de son soutien, des données des essais fonctionnels, des échéances de livraison des versions logicielles et des données sur les caractéristiques du produit ou du service sur le terrain permette de faire un suivi des exigences stipulées dans le plan global de sûreté de fonctionnement. Il convient que des enregistrements appropriés relatifs aux activités logicielles soient maintenus parallèlement à ceux relatifs aux autres activités du projet afin de faciliter l'identification de l'origine des problèmes et la corrélation des données pour cerner la cause première des problèmes.

6.2.4 Gestion de configuration

Il est recommandé que la gestion de configuration soit conforme à 6.1.4 de la CEI 60300-2.

Il est recommandé d'établir et de mettre en oeuvre un plan de gestion de configuration pour le logiciel. Ce plan est utilisé dans tout le projet pour identifier, contrôler, prendre en compte l'état, évaluer, gérer les modifications, gérer la diffusion et livrer le logiciel utilisé.

Il est conseillé que la gestion de configuration du logiciel soit un produit fonctionnel livrable, faisant partie du plan de gestion de configuration, relatif au projet global, du système ou du produit.

6.3 Revue de contrat et liaison

6.3.1 Revue de contrat

Il est recommandé que la revue de contrat soit conforme à 6.2.1 de la CEI 60300-2.

Il convient d'effectuer des revues sur les problèmes spécifiques liés au logiciel concerné par le contrat en même temps que la revue du projet global. Les exigences spécifiques aux contrats concernant les produits logiciels livrables sont revues avec le client pour acceptation, et le cas échéant, avec les fournisseurs chargés par contrat de sous-produits. Lorsque des divergences apparaissent, il est recommandé que les problèmes spécifiques soient résolus et que le contrat soit amendé en fonction du dernier état. Il convient de conserver les minutes des revues de contrats.

6.3.2 Représentants de la direction

IL convient que les représentants de la direction soient choisis conformément à 6.2.2 de la CEI 60300-2.

Les fonctions et responsabilités des représentants de la direction s'organisent à deux niveaux: au niveau de l'organisation et au niveau du projet.

6.2.2 Project decision management

Project decision management should be conducted in accordance with 6.1.2 of IEC 60300-2.

The management of a software project should be an integral part of the overall project management for the system or product containing the software. Milestone objectives related to software should reflect a coordinated set of deliverables at scheduled targets to facilitate project decision at management reviews.

6.2.3 Traceability management

Traceability management should be in accordance with 6.1.3 of IEC 60300-2.

Traceability of software development and support efforts, functional test data, software release schedules, and field performance data of the product or service deliverables should reflect the requirements stipulated in the overall dependability plan. Relevant records pertaining to software activities should be maintained along with other project activities to facilitate source identification and data correlation for tracking root cause problems.

6.2.4 Configuration management

Configuration management should be in accordance with 6.1.4 of IEC 60300-2.

A configuration management plan should be established and implemented for the software project. This plan is used for identification, control, status accounting, evaluation, change management, release management and delivery of the software involved in the overall project.

Software configuration management should be a functional deliverable item as part of the system or product configuration management plan of the overall project.

6.3 Contract review and liaison

6.3.1 Contract review

Contract review should be in accordance with 6.2.1 of IEC 60300-2.

Specific issues concerning the software involved in a contract should be reviewed in conjunction with the overall project review process. Specific contract requirements pertaining to the software deliverables are reviewed with the customer for acceptance and, where applicable, with the suppliers of subcontract items. Where discrepancies occur, the specific issues should be resolved and the contract amended to reflect the latest status. Contract review records should be maintained.

6.3.2 Management representative

Management representatives should be selected in accordance with 6.2.2 of IEC 60300-2.

There are two aspects related to the function and responsibility of the management representative; one is at the organizational level, the other is at the project level.

Au niveau de l'organisation, le représentant de la direction est responsable de l'application du système qualité en conformité avec les normes d'assurance de la qualité de l'ISO 9000. La sûreté du fonctionnement, en tant que discipline technique spécifique, contribue à l'organisation en termes de valeur ajoutée des principes et pratiques de gestion de la sûreté de fonctionnement. L'effort consacré à la sûreté du fonctionnement au sein de l'organisation permet d'accroître l'efficacité globale du système qualité. Si nécessaire, le représentant de la direction peut trouver une aide technique auprès d'autres spécialistes de la qualité et de la sûreté de fonctionnement afin de résoudre les problèmes généraux et systémiques concernant le système qualité au niveau de l'organisation. Généralement, au sein de l'organisation, le représentant de la direction assume la fonction d'administrateur, de directeur ou de responsable de la qualité.

Au niveau du contrat ou du projet, le représentant de la direction traite des problèmes spécifiques relatifs à la livraison contractuelle du produit ou système. Le représentant de la direction est chargé des relations avec le client et les fournisseurs d'un projet spécifique du point de vue de l'assurance de la qualité et de la sûreté de fonctionnement des produits livrables. Il convient que le représentant de la direction au niveau du projet ait les connaissances appropriées concernant le projet spécifique. Si nécessaire, il peut chercher le soutien d'experts pour résoudre les problèmes d'un projet ou contrat. En général, le représentant de la direction au niveau d'un projet assume la fonction de chef de projet, d'ingénieur principal ou d'ingénieur en chef.

6.4 Exigences concernant la sûreté de fonctionnement

6.4.1 Spécification des exigences de sûreté de fonctionnement

Il est recommandé que la spécification des exigences de sûreté de fonctionnement soit conforme à 6.3.1 de la CEI 60300-2.

Il est recommandé que la spécification des exigences concernant la sûreté de fonctionnement comprenne:

– les conditions de fonctionnement du système ou du produit qui comportent le logiciel et qui influent sur les exigences de fonctionnement du logiciel;

– le caractère critique du logiciel dans le système ou le produit pour réaliser l'objectif global de disponibilité;

– la durée du temps d'indisponibilité autorisée et la fréquence d'interruption de service allouée ou due aux problèmes de logiciel en relation avec l'indisponibilité globale du système;

– les exigences concernant le taux de couverture des essais du logiciel et concernant les diagnostics logiciels;

– les exigences concernant la qualification du logiciel et sa réception;

– les exigences concernant l'essai du logiciel et son intégration au système;

– les exigences concernant la documentation du logiciel;

– les exigences concernant la gestion de configuration du logiciel;

– les exigences concernant la livraison et la mise à jour du logiciel;

– les exigences concernant la maintenance du logiciel.

6.4.2 Interprétation des exigences

Il est recommandé que l'interprétation des exigences concernant la sûreté de fonctionnement du logiciel soit conforme à 6.3.2 de la CEI 60300-2.

The management representative at the organizational level is responsible for the quality system implementation within the organization. This is in conformance with the ISO 9000 quality management standards. Dependability as a specific technical discipline contributes to the organizational infrastructure in terms of value added dependability management principles and practices. The dependability effort within the organization enhances the overall effectiveness of the quality system. Where necessary, the management representative may seek technical assistance from other quality and dependability specialists for resolution of generic or systemic issues related to the quality system at the organization level. A typical position in an organization for the management representative is the quality director, manager, or administrator.

The management representative at the contract or project level deals with the specific issues related to contract delivery of the product or system. The management representative is responsible for interfacing with the customer and suppliers of a specific project by ensuring the quality and dependability of the deliverables. The management representative at the project level should have adequate knowledge concerning the specific project involved. Where necessary, technical experts may be sought to resolve project or contract related problems. A typical position for the management representative in a project is the project manager, principal or lead engineer.

6.4 Dependability requirements

6.4.1 Specification of dependability requirements

Specification of dependability requirements should be in accordance with 6.3.1 of IEC 60300-2.

Specification of dependability requirements for software should consider:

– operating environment of the system or product containing the software and affecting the functional requirements of the software;
– the criticality of the software application in the system or product to achieve the overall availability performance objective;
– permissible downtime duration and outage frequency allocated or contributed by software problems in relation to the overall system downtime;
– software diagnostic and test coverage requirements;
– software qualification and acceptance requirements;
– software test and system integration requirements;
– software documentation requirements;
– software configuration management requirements;
– software release and update requirements;
– software maintenance requirements.

6.4.2 Requirements interpretation

Requirements interpretation should be in accordance with 6.3.2 of IEC 60300-2.

Il convient de distinguer deux types d'exigences:

a) exigences affectant les caractéristiques du système ou du produit global;

b) exigences relatives à l'utilisation ou à l'application prévue du logiciel.

La liste des exigences concernant la sûreté de fonctionnement indiquée en 6.4.1 n'est pas exhaustive. Il se peut qu'il y ait des différences dans l'interprétation des exigences lorsque le même logiciel de base est utilisé dans un système ou produit pour des applications différentes, ou pour un fonctionnement dans des conditions différentes.

6.4.3 Répartition des exigences

Il est recommandé que la répartition des exigences soit conforme à 6.3.3 de la CEI 60300-2.

Il convient de mettre l'accent sur la répartition des exigences concernant la sûreté du fonctionnement conformément à la structure du système. L'application des exigences concernant la sûreté du fonctionnement sur le système comprenant matériel et logiciel est un processus clé. Cette approche hiérarchisée permet de répartir les ressources en fonction des exigences de l'ensemble du système. Elle facilite le compromis sur la conception fonctionnelle. Elle permet de rationaliser les décisions d'achat et de vente, et aussi de planifier, de mettre en oeuvre le travail approprié ou le degré de rigueur technique exigé pour le développement, l'acquisition ou le soutien du logiciel à différents niveaux d'applications critiques.

6.5 Ingénierie

6.5.1 Ingénierie de la fiabilité

Il est recommandé que l'ingénierie de la fiabilité soit conforme à 6.4.1 de la CEI 60300-2.

Le travail d'ingénierie de la fiabilité relatif au logiciel est associé au degré de rigueur d'ingénierie lors de l'application des méthodes appropriées.

La part de fiabilité du logiciel dans le système ou le produit comportant le logiciel dépend fortement du processus de développement du logiciel.

6.5.2 Ingénierie de la maintenabilité

Il est recommandé que l'ingénierie de la maintenabilité soit conforme à 6.4.2 de la CEI 60300-2.

Le travail d'ingénierie de la maintenabilité relatif au logiciel est associé au degré de rigueur lors de l'application des méthodes appropriées.

6.5.3 Ingénierie de la logistique de maintenance

Il est recommandé que l'ingénierie de la logistique de maintenance soit conforme à 6.4.3 de la CEI 60300-2.

Il convient de noter que tout le travail de maintenance effectué sur le logiciel est en rapport avec les erreurs découvertes dans la conception du logiciel, avec les changements des exigences d'application du logiciel ou avec la maintenance perfective (amélioration du logiciel) afin de limiter les insuffisances de réalisation du logiciel plutôt que de réagir à l'apparition d'une défaillance du système.

6.5.4 Ingénierie de la testabilité

Il est recommandé que l'ingénierie de la testabilité soit conforme à 6.4.4 de la CEI 60300-2.

Le travail d'ingénierie visant à assurer la testabilité du logiciel doit comporter:

The requirements interpretation should distinguish:

a) those requirements affecting the overall system or product performance;

b) those requirements related to the intended use or application of the software.

The list of dependability requirements provided in 6.4.1 is not exhaustive. There may also be differences in the requirements interpretation when the same basic software is used in a system or product for different applications or operating in separate environments.

6.4.3 Requirements allocation

Requirements allocation should be in accordance with 6.3.3 of IEC 60300-2.

Emphasis should be placed on the allocation of dependability requirements in accordance with the system structure. Mapping of the dependability of requirements onto the system architecture consisting of hardware and software components is a key process. This top down approach provides a means to allocate proper resources relevant to the total system requirements. It facilitates functional design trade-off, permits rationalization of make/buy decisions, and allows planning and implementation of the appropriate level of effort or the degree of engineering rigour required in the development, acquisition, or support of the software for various levels of critical applications.

6.5 Engineering

6.5.1 Reliability engineering

Reliability engineering should be in accordance with 6.4.1 of IEC 60300-2.

Reliability engineering effort applicable to software is associated with the degree of engineering rigour in the application of relevant methods.

Reliability contribution of software components to a system or product containing the software is highly dependent on the development process and the design of the software.

6.5.2 Maintainability engineering

Maintainability engineering should be in accordance with 6.4.2 of IEC 60300-2.

Maintainability engineering effort applicable to software is associated with the degree of engineering rigour in the application of relevant methods.

6.5.3 Maintenance support engineering

Maintenance support engineering should be in accordance with 6.4.3 of IEC 60300-2.

It should be noted that all software maintenance effort is in response to discovered errors in the software design, changes to the software application requirements, or the perfective maintenance of the software (software enhancement) to reduce a shortcoming in the software implementation rather than reacting to a system failure occurrence.

6.5.4 Testability engineering

Testability engineering should be in accordance with 6.4.4 of IEC 60300-2.

The engineering effort to ensure software testability includes:

– les spécifications de conception pour la testabilité;
– les normes et méthodes d'essai;
– la couverture des essais du logiciel;
– la couverture des essais des exigences relatives au système;
– l'intégration du logiciel et les procédures d'essai de conformité.

La testabilité est la limite dans laquelle un essai objectif et réalisable peut être effectué au niveau du logiciel dans le but de savoir si une prescription est ou n'est pas satisfaite. La couverture des essais est la limite dans laquelle les cas d'essais réalisés sur le système ou sur le logiciel permettent de contrôler leur conformité avec les exigences établies.

Il convient de noter que le but des essais effectués pendant le développement du logiciel consiste à trouver les pannes des composants du logiciel. Le but de l'essai du diagnostic d'erreurs pendant la maintenance consiste à déterminer l'origine de l'erreur ou du mauvais fonctionnement identifié au niveau du système.

6.5.5 Ingénierie des facteurs humains

Il est recommandé que l'ingénierie des facteurs humains soit conforme à 6.4.5 de la CEI 60300-2.

Les facteurs humains ont un impact important sur les caractéristiques du système. Le travail d'ingénierie des facteurs humains est directement lié à la planification, à la conception et à l'exécution du logiciel utilisé pendant le fonctionnement du système. Des lignes directrices et des normes de conception sont nécessaires pour que la conception du logiciel soit cohérente afin de faciliter la testabilité et l'intégration. Il convient d'élargir le champ d'application des cas d'essais et des méthodes d'essai documentés afin d'inclure les éléments des facteurs humains relatifs au fonctionnement et à la maintenance du logiciel et afin que toutes les exigences du système concernant la sûreté du fonctionnement soient satisfaites.

En fonction de la criticité de l'application du système, il convient que le travail d'ingénierie des facteurs humains exigé soit cohérent avec l'application du projet. Il convient d'étudier l'impact potentiel sur l'environnement immédiat d'un mauvais fonctionnement du système dû à une erreur humaine. L'apport du facteur humain au niveau du logiciel est associé au degré de rigueur technique lors de l'application des méthodes appropriées.

6.6 Produits fournis par des tiers

6.6.1 Produits fournis par des sous-traitants

Il convient de considérer les exigences concernant les produits fournis par des sous-traitants conformément à 6.5.1 de la CEI 60300-2.

Les questions suivantes peuvent être à considérer dans le cas de produits fournis par des sous-traitants à intégrer dans le système d'accueil:

– sous-traitance complète d'un composant logiciel ou d'un sous-système;
– acquisition ou approvisionnement des progiciels disponibles en stock;
– sous-traitance en vue de modifier le logiciel existant.

Il convient, si possible, d'inclure dans le contrat de sous-traitance les exigences concernant la sûreté de fonctionnement du système d'accueil. Il est recommandé d'inclure dans les plans de sous-traitance les dates et étapes de livraisons, le contrôle du fournisseur, les revues de contrat, la documentation, les critères de réception et les exigences concernant la logistique de maintenance du logiciel.

– design specifications for testability;
– test methods and standards;
– test coverage of the software component;
– test coverage of system requirements;
– software integration and testing for conformance.

Testability is the extent to which an objective and feasible test can be designed into the software to determine whether a requirement is met. Test coverage is the extent to which the test cases are developed to test the system or the software components for conformance to established requirements.

It should be noted that the purpose of testing during software development is to find faults within the software components. The purpose of diagnostic testing during software maintenance is to determine the root cause of an identified system failure or malfunction.

6.5.5 Human factors engineering

Human factors engineering should be in accordance with 6.4.5 of IEC 60300-2.

Human factors have significant impact on system performance. The level of engineering effort relates directly to the planning, design and execution of the software involved in system operation. Design guidelines and standards should be used to ensure consistency in software design to facilitate testability and integration. Documented test cases and test procedures should be extended to include human factor elements relating to the operation and maintenance of the software to ensure that the overall dependability requirements of the system are met.

Depending on the criticality of system application, the level of human engineering effort required should be consistent with the project application. The potential impact on its immediate environment in case of a system malfunction due to human error should be explored. Human engineering effort applicable to software is associated with the degree of engineering rigour in the application of relevant methods.

6.6 Externally provided products

6.6.1 Subcontracted products

Requirements dealing with subcontracting and subcontracted products should be considered in accordance with 6.5.1 of IEC 60300-2.

The following issues may be applicable to subcontracted software products for integration into the host system:

– subcontracting complete development of the software component or subsystem;
– acquisition or out-sourcing of off-the-shelf software packages;
– subcontracting for modification of existing software.

Where applicable, the dependability requirements of the host system should be reflected in the subcontract statement of work. Subcontract plans should include schedule and milestone deliverables, supplier monitoring, contract reviews, documentation, acceptance criteria and software maintenance support requirements.

6.6.2 Produits fournis par le client

Il convient de considérer les produits fournis par le client conformément à 6.5.2 de la CEI 60300-2.

Les logiciels fournis par le client peuvent être des logiciels existants ou un sous-système destiné à être intégré au système d'accueil et à fonctionner au sein de celui-ci dans le cadre d'un contrat de livraison. Il convient de considérer les exigences suivantes:

– spécifications du logiciel ou du sous-système fourni par le client;
– exigences concernant les interfaces;
– exigences concernant l'intégration et les essais;
– une procédure de revue en cas de défaillance du système d'accueil due à une erreur du logiciel fourni par le client;
– exigences concernant la gestion de la configuration.

6.7 Analyse, prévision et revue de conception

6.7.1 Généralités

La méthodologie d'analyse du logiciel est en général basée sur l'expérience pratique et sur les données d'essai relatives aux applications spécifiques de logiciel et aux environnements d'exploitation associés. Les modèles de performance du logiciel, y compris les modèles d'émulation de fiabilité des systèmes de logiciel, sont mis à disposition pour prévoir la fiabilité et évaluer sa croissance. Ces modèles représentent les fonctions mathématiques relatives aux paramètres de performance du logiciel spécifique afin de fournir un résultat quantitatif à partir des données d'expériences. C'est la raison pour laquelle ces modèles de performance des logiciels sont spécifiques à une application.

Il n'existe pas de modèles généraux ni de normes concernant l'analyse et l'évaluation de logiciels qui respectent toutes les exigences d'application et toutes les situations de contrat. Toutefois, il existe dans l'industrie de nombreux codes de bonne pratiqe développés spécifiquement pour l'analyse des logiciels, tels que:

– analyse de la complexité du logiciel pour estimer le nombre d'erreurs dans un ensemble donné de modules logiciels;
– analyse de la couverture du code pour déterminer la complétude de l'essai;
– corrélation avec la classification de défauts des logiciels en vue d'une analyse rapide des causes et de l'amélioration du logiciel pendant son développement.

Les articles suivants présentent les méthodes normalisées identifiées dans la CEI 60300-2.

6.7.2 Analyse des modes de défaillance et de leurs effets

Il convient, si possible, d'effectuer une analyse des modes de défaillance et de leurs effets (AMDE) [4] conformément à 6.6.1 de la CEI 60300-2.

Il convient que l'analyse des modes de défaillance et de leurs effets (AMDE) applicable au logiciel soit traitée depuis le niveau de l'architecture de la conception jusqu'au niveau fonctionnel approprié pour les applications critiques. Au niveau fonctionnel le logiciel est considéré comme une boîte noire. Il convient d'analyser les effets d'une défaillance de logiciel afin d'en évaluer les risques sur le fonctionnement du système. Il convient, si possible, de procéder à une évaluation quantitative afin de déterminer les conséquences de la défaillance sur les choix de conception et l'amélioration du système.

6.6.2 Customer-provided products

Customer-provided products should be considered in accordance with 6.5.2 of IEC 60300-2.

Customer-provided software products may be existing software products or a subsystem needed for integration and operation with the host system under contract for delivery. The following requirements should be considered:

– specifications of the customer-provided software product or subsystem;
– interface requirements;
– integration and test requirements;
– a review process in case of host system failures traceable to customer-provided software faults;
– configuration management requirements.

6.7 Analysis, prediction and design review

6.7.1 General

Software analysis methodology in general is based on practical experience and test data with specific software applications and associated operating environments. Software performance models, including those that emulate reliability performance of software systems, are being formulated for reliability prediction and reliability growth assessment purposes. These models represent the mathematical functions relating to specific software performance parameters to provide a quantitative output using the engineering data input. Hence these software performance models are application specific.

There are no generic models or standards for software analysis and evaluation that will meet all application requirements and contract situations. However, there are numerous industry best practices developed for specific software analysis such as:

– software complexity analysis to estimate the fault content in a given set of software modules;
– analysis of code coverage to determine test completeness;
– correlation of software defects classification for rapid root cause analysis and in-process improvement.

The following subclauses present the standard methods identified in IEC 60300-2.

6.7.2 Fault mode and effects analysis

Fault mode and effects analysis (FMEA) [4] should be conducted, where applicable, in accordance with 6.6.1 of IEC 60300-2.

FMEA [4] applicable to software systems should be dealt with at the architectural design level and cascaded down to the functional level as appropriate for critical applications. At the functional level the software component is treated as a black box. The effect of a software fault should be analysed to determine its criticality to the system operation. Where applicable, quantitative assessment should be conducted to determine the failure impact for design trade-off and system performance improvement.

Exemples possibles de modes de défaillance de logiciels:

– une mauvaise sortie apparaît pour une entrée correcte;
– une entrée incorrecte n'est pas reconnue;
– le logiciel est altéré et indique une erreur fonctionnelle significative;
– une boucle infinie apparaît et aucune sortie n'est fournie par la fonction du logiciel;
– aucune sortie n'est fournie dans l'intervalle de temps exigé.

6.7.3 Analyse par arbre de panne

Il convient, si possible, d'effectuer l'analyse par arbre de panne (AAP) [5] conformément à 6.6.2 de la CEI 60300-2.

L'AAP est applicable au logiciel au niveau fonctionnel dans les cas où les performances de sûreté de fonctionnement du logiciel à l'intérieur du système global peuvent être évaluées. L'AAP est une approche descendante du système visant à déterminer si la probabilité de défaillance d'un sous-système comprenant le composant logiciel est critique.

L'AAP pourrait aussi être réalisée, selon le cas, en même temps que d'autres techniques d'analyse, afin de déterminer la fiabilité de la fonction du logiciel lors de l'exploitation de systèmes critiques. Les résultats de l'AAP et de l'AMDE peuvent être qualitatifs ou quantitatifs, ou les deux à la fois. Toutefois, il convient de noter que la plupart des analyses AMDE et AAP effectuées jusqu'à présent sont qualitatives.

6.7.4 Analyse des contraintes et des charges

Il convient, si possible, d'effectuer l'analyse des contraintes et des charges conformément à 6.6.3 de la CEI 60300-2.

Dans le cadre de l'application au logiciel, l'analyse des contraintes et des charges est liée à la vitesse et à l'aptitude du logiciel à traiter un débit élevé d'informations. Il n'existe aucune méthode ni norme spécifique concernant l'analyse des contraintes et des charges pour les logiciels.

6.7.5 Analyse des facteurs humains

Il convient, si possible, d'effectuer l'analyse des facteurs humains conformément à 6.6.4 de la CEI 60300-2.

L'analyse des facteurs humains est une discipline technique qui détermine les conséquences des erreurs humaines sur la conception, l'analyse, l'exécution et la maintenance du logiciel ou du système. Aucune norme spécifique sur l'analyse des facteurs humains pour les logiciels n'a été établie. Cependant, en appliquant l'AMDE, l'AAP, une analyse du risque et d'autres techniques applicables, les facteurs humains pourraient être considérés dans l'évaluation des performances de sûreté de fonctionnement du système comportant un logiciel.

6.7.6 Prévisions

Il convient, si possible, d'effectuer les prévisions conformément à 6.6.5 de la CEI 60300-2.

Il est recommandé que les prévisions liées aux produits logiciels prennent en compte l'environnement de l'application, la charge et la complexité, l'architecture de la configuration du système ainsi que les données empiriques utilisées pour déterminer les prévisions de la fiabilité des produits logiciels.

Examples of software fault modes may include:

– a wrong output is provided for a correct input;
– an incorrect input is not recognized;
– the software is corrupted indicating a significant functional error;
– an infinite loop occurs and no output at all is provided by the software function;
– no output is supplied within the required time interval.

6.7.3 Fault tree analysis

Fault tree analysis (FTA) [5] should be conducted, where applicable, in accordance with 6.6.2 of IEC 60300-2.

FTA is applicable to software at the functional level where the dependency of the software component in the overall system performance may be assessed. FTA is a top down system approach to determine whether the probability of a lower system branch containing the software component is critical.

FTA could also be conducted in conjunction with other analysis techniques as appropriate for determining the reliability performance of the software function for operation of critical systems. The results of FTA and FMEA could be qualitative or quantitative or both. However, it should be noted that most FMEA and FTA performed to date on software are qualitative.

6.7.4 Stress and load analysis

Stress and load analysis should be conducted, where applicable, in accordance with 6.6.3 of IEC 60300-2.

In the context of software application, stress and load analysis is associated with the speed and capacity for the software function to process information throughput. There are no specific methods or standards established for stress and load analysis for software.

6.7.5 Human factors analysis

Human factors analysis should be conducted, where applicable, in accordance with 6.6.4 of IEC 60300-2.

Human factors analysis is a technical discipline which determines the effects of human errors affecting the design, analysis, execution and maintenance of the software product or system. There are no specific standards established for human factors analysis for software. However, in the application of FMEA, FTA, risk analysis and other applicable techniques, the human factors element could be considered as input for the assessment of dependability performance of systems containing software.

6.7.6 Predictions

Predictions should be conducted, where applicable, in accordance with 6.6.5 of IEC 60300-2.

Predictions associated with software products should consider the application environment, operating load and complexity, architecture of system configuration, and the empirical data used to base the reliability performance predictions of the software products.

Les prévisions pour les logiciels peuvent être effectuées selon trois approches différentes. La première est fondée sur les propriétés du processus de développement du système. La seconde est fondée sur les caractéristiques du produit logiciel. La troisième est fondée sur les données empiriques rassemblées à partir de processus de vérification et du fonctionnement réel du logiciel.

Les modèles de prévision tirés des propriétés du processus de développement du logiciel sont influencés par les paramètres du processus. Le principe est le suivant: les disciplines de la gestion utilisées pour le développement du logiciel seraient susceptibles de fournir des objectifs de fiabilité pour le logiciel. A cet égard, les paramètres du processus sont utilisés comme points de repère pour améliorer la fiabilité.

Les modèles de prévision tirés des caractéristiques du produit logiciel sont influencés par les paramètres du produit logiciel tels que le type, la structure et la complexité du logiciel. En général, la prévision de la fiabilité basée sur de tels modèles est utilisée pour l'évaluation des progiciels et l'analyse comparative.

Les modèles de prévision tirés des données de performance du logiciel sont influencés par l'application spécifique et l'environnement d'exploitation du logiciel. Des méthodes statistiques sont utilisées pour la prévision de la fiabilité afin d'évaluer l'évolution de la fiabilité à partir des données observées.

Les méthodes de prévisions concernant les logiciels évoluent sans cesse. Les méthodologies et les modèles existants sont très spécifiques au produit et différentes selon les applications. On ne dispose pas de méthode générale pour évaluer les prévisions de la fiabilité des logiciels. A ce stade, la part de fiabilité apportée par les logiciels à l'intérieur d'un système est la plupart du temps évaluée à l'aide des données observées du système contenant le logiciel.

6.7.7 Analyse de compromis

Il convient, si possible, d'effectuer l'analyse de compromis conformément à 6.6.6 de la CEI 60300-2.

En considérant les composants du logiciel comme des fonctions intégrées à un système ou à un produit comportant un logiciel, des méthodes conventionnelles telles que l'AMDE et l'AAP pourraient en effet être utilisées pour l'analyse de compromis lors de la conception, des décisions d'achat et de vente, et de l'analyse comparative des différentes solutions. L'analyse de compromis est utilisée lors de la prise de décision pour sélectionner une approche appropriée du logiciel, une approche matérielle alternative, ou une approche de l'architecture de conception combinant matériel et logiciel afin de réaliser les caractéristiques du système en satisfaisant aux exigences de rentabilité du projet.

6.7.8 Analyse des risques

Il convient, si possible, d'effectuer l'analyse des risques conformément à 6.6.7 de la CEI 60300-2.

Pour effectuer l'analyse des risques, il convient d'utiliser comme référence la CEI 60300-3-9 [6].

6.7.9 Revue de conception formalisée

Il convient d'effectuer les revues de conception formalisées conformément à 6.6.8 de la CEI 60300-2.

Pour effectuer une revue de conception formalisée, il convient d'utiliser la CEI 61160 comme ligne directrice.

There are three generic approaches towards prediction methods for software. The first is based on the software development process properties. The second is based on the software product characteristics. The third is based on empirical data gathered from verification processes and actual operation of the software.

Prediction models derived from software development process properties are influenced by the process parameters. The concept is that the management disciplines used for software development could provide reliability projection targets for the software. In this respect, the process parameters are used as benchmarks for reliability improvement.

Prediction models derived from software product characteristics are influenced by the software product parameters such as form, structure and complexity of the software. Reliability prediction based on such models is generally used for off-the-shelf software product evaluation and comparative analysis.

Prediction models derived from software performance data are influenced by the specific application and operating environment of the software. Statistical methods are used for reliability prediction to estimate reliability growth projection based on observed data.

Prediction methods for software are still evolving. Existing methodologies and models are very much product specific and application oriented. There are no standard methods available for generic application of reliability prediction for software. At this point, reliability contribution from software components within a system is mostly estimated, based on observed system performance data containing the software.

6.7.7 Trade-off analysis

Trade-off analysis should be conducted, where applicable, in accordance with 6.6.6 of IEC 60300-2.

By treating software components as functions within a system or product containing software, conventional methods such as FMEA and FTA could be effectively used for design trade-off, make/buy decisions, and comparative analysis for alternate solutions. Trade-off analysis is used for decision making to select an appropriate software approach, an alternate hardware approach, or a combined hardware and software solution in the design architecture to achieve system performance to meet cost-effective project requirements.

6.7.8 Risk analysis

Risk analysis should be conducted, where applicable, in accordance with 6.6.7 of IEC 60300-2.

IEC 60300-3-9 [6] should be used as a reference for conducting risk analysis.

6.7.9 Formal design review

Formal design review should be conducted in accordance with 6.6.8 of IEC 60300-2.

IEC 61160 should be used as guidance to conduct formal design review.

6.8 Vérification, validation et essai

6.8.1 Planification des vérifications, des validations et des essais

Il convient d'effectuer la planification des vérifications, des validations et des essais conformément à 6.7.1 de la CEI 60300-2.

La vérification consiste à confirmer à l'aide d'un examen et de preuves objectives que les exigences spécifiées ont été satisfaites. Pour la conception et le développement des logiciels, la vérification consiste à contrôler le résultat d'une activité donnée afin de voir s'il est conforme à la prescription donnée pour cette activité.

Il est conseillé que les activités liées à la vérification des logiciels comportent:

– une vérification des contrats;
– une vérification du processus;
– une vérification des exigences;
– une vérification de la conception;
– une vérification du code;
– une vérification de l'intégration;
– une vérification de la documentation.

La validation consiste à confirmer à l'aide d'un examen et de preuves objectives que les exigences spécifiques à une utilisation sont satisfaites. Normalement, la validation est effectuée sur le produit final dans des conditions d'exploitation définies.

Pour les logiciels, il est important de confirmer que les besoins et les attentes sur le produit sont compris. En ce qui concerne les applications à haut risque où des ressources importantes sont engagées sur le logiciel, il est essentiel que des actions nécessaires soient entreprises pour satisfaire les attentes de l'utilisateur du logiciel ou du système comportant un logiciel. Des exemples de telles actions sont:

– l'utilisation de simulations pour comprendre quels sont les besoins en termes de performance;
– l'utilisation de prototypes pour mettre au clair les interactions existant entre l'utilisateur et le logiciel;
– l'utilisation de modèles formalisés pour clarifier la fonctionnalité du logiciel proposé.

Il convient que les activités liées à la validation des logiciels comportent:

– une préparation des exigences d'essai sélectionnées, des spécifications d'essais et des jeux d'essais pour l'analyse des résultats;
– des exigences d'essai reflétant l'utilisation prévue du logiciel;
– des essais;
– une analyse des résultats des essais pour voir s'ils sont conformes.

L'analyse et la revue des logiciels incluent des techniques telles que l'inspection de code, le cheminement pas à pas, les descriptions formelles, les exécutions symboliques, la démonstration du programme, etc. Les essais de vérification et de validation du logiciel incluent des techniques telles que, les «tests boîte noire», les «tests boîte de verre», les essais en charge, les tests statistiques, les tests de croissance de la fiabilité, etc.

La planification des essais comporte toute les actions de vérification et de validation concernant le logiciel spécifique dans le cadre du contrat. Il convient d'intégrer la planification des essais à la planification de l'ensemble du projet.

6.8 Verification, validation and test

6.8.1 Verification, validation and test planning

Verification, validation and test planning should be conducted in accordance with 6.7.1 of IEC 60300-2.

Verification is the confirmation by examination and provision of objective evidence that specified requirements have been fulfilled. In design and development of software, verification concerns the process of examining the result of a given activity to determine conformity with the stated requirement for that activity.

Activities associated with the verification of software should include:

– contract verification;
– process verification;
– requirements verification;
– design verification;
– code verification;
– integration verification;
– documentation verification.

Validation is the confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled. Validation is normally performed on the final product under defined operating conditions.

For software it is important to confirm that the needs and expectations for the software are understood. For high risk applications where significant resources are being expended on the software, it is essential that necessary actions be taken to validate the user expectations of the software or the system containing software. Some examples of such actions are:

– the use of simulations to understand performance expectations;
– the use of prototypes to clarify the user interactions with the software;
– the use of formal models to clarify the functionality of the proposed software.

Activities associated with the validation of software should include:

– preparation of selected test requirements, test specifications and test cases for analysis of test results;
– ensuring test requirements reflect the intended use of the software;
– testing;
– analysis of test results for conformity.

Examples of software analysis and review include techniques such as code inspection, walk through, formalised descriptions, symbolic executions, programme proving, etc. Examples of software verification and validation testing include techniques such as black box, white box, load testing, statistical testing, reliability growth testing, etc.

Test planning includes all applicable verification and validation activities for the specific software under contract. Test planning should be part of the overall project plan.

6.8.2 Essai de durée de vie

L'essai de durée de vie est une activité qui concerne le matériel et qui n'est pas applicable au logiciel en tant que tel. Cependant, l'essai de durée de vie est applicable pour évaluer la durée de vie des produits comportant à la fois du matériel et du logiciel lorsque ce dernier est destiné à servir de composant fonctionnel.

6.8.3 Essai de sûreté de fonctionnement

Il convient d'effectuer l'essai de sûreté de fonctionnement conformément à 6.7.3 de la CEI 60300-2.

Il convient, si possible, de réaliser les essais relatifs à l'évaluation, la démonstration et la réception de la disponibilité du système comportant un logiciel en même temps que les autres activités d'essai. Il est recommandé que les données rassemblées fournissent des informations appropriées pour l'analyse afin de déterminer la disponibilité du système concernant la réception de la sûreté de fonctionnement.

6.8.4 Essai de croissance de fiabilité

Il convient d'effectuer l'essai de croissance de fiabilité conformément à 6.7.4 de la CEI 60300-2.

La CEI 61014 [7] donne des lignes directrices pour le développement de programmes et de procédures de croissance de fiabilité. La CEI 61164 [8] donne une méthodologie pour les essais et l'estimation de la croissance de fiabilité.

Les modèles spécifiques sur la croissance de fiabilité applicables au logiciel comportent les éléments suivants:

– une représentation du processus de défaillance à l'aide d'un ensemble de formules mathématiques comprenant certains paramètres;
– une méthode d'évaluation des paramètres à l'aide d'une analyse des données relatives à des défaillances antérieures;
– une méthode pour combiner les valeurs de paramètres évaluées avec les formules afin d'obtenir des évaluations numériques des mesures de fiabilité.

6.8.5 Essai en production

Il convient d'effectuer l'essai en production conformément à 6.7.5 de la CEI 60300-2.

En général, l'essai est applicable aux matériels et aux systèmes comportant à la fois du matériel et un logiciel intégrés au processus de fabrication. Les essais en production ne s'appliquent pas aux essais de logiciels en tant que tels.

6.8.6 Essai d'acceptation

Il convient d'effectuer l'essai d'acceptation conformément à 6.7.6 de la CEI 60300-2.

L'essai d'acceptation est lié à la vérification et à la validation du logiciel. Il existe trois niveaux d'essai d'acceptation pour les logiciels:

a) soumettre à l'essai chaque unité ou chaque composant du logiciel afin de s'assurer que cette unité ou ce composant est conforme aux spécifications ou normes établies;
b) soumettre à l'essai les unités et composants logiciels intégrés comme agrégat; cet essai est généralement connu sous le nom d'essai d'intégration;
c) soumettre à l'essai l'installation du logiciel pour acceptation finale et mise en service afin d'être sûr que le logiciel fonctionne dans le système configuré pour un environnement réel et dans les conditions établies spécifiées dans le contrat.

6.8.2 Life testing

Life testing is a hardware activity and not applicable to software as a stand alone entity. However, life testing is applicable for life evaluation of products containing combined hardware and software where the software is incorporated to serve as a functional component.

6.8.3 Dependability testing

Dependability testing should be conducted in accordance with 6.7.3 of IEC 60300-2.

Testing related to the evaluation, demonstration and acceptance of the availability performance of the system containing software should be conducted in conjunction with other test activities, where applicable to the project. The data collected should provide adequate information for analysis to determine the availability performance of the system for dependability acceptance.

6.8.4 Reliability growth testing

Reliability growth testing should be conducted in accordance with 6.7.4 of IEC 60300-2.

IEC 61014 [7] provides guidance for development of reliability growth programmes and procedures. IEC 61164 [8] provides methodology for reliability growth testing and estimation.

Specific reliability growth models applicable to software consist of the following elements:

– a representation of the failure process by a set of mathematical formulae incorporating certain parameters;

– a method of estimating the parameters by analysis of previous failure data;

– a method of combining the estimated parametric values with the formulae to obtain numerical estimates of reliability measures.

6.8.5 Production testing

Production testing should be conducted in accordance with 6.7.5 of IEC 60300-2.

Production testing is usually applicable to hardware products and to systems containing both hardware and software components as part of a manufacturing process. Production testing is not applicable to testing software as a stand alone item.

6.8.6 Acceptance testing

Acceptance testing should be conducted in accordance with 6.7.6 of IEC 60300-2.

Acceptance testing is associated with software verification and validation. There are three levels of testing of software for acceptance:

a) testing of each software unit or component to ensure conformance to established specifications or standards;

b) testing of integrated software units and components as an aggregate; this is generally known as integration testing;

c) testing of the software installation for final acceptance and commissioning to ensure that the software works in the system configured to operate in actual environment and established conditions as specified in the contract.

6.8.7 Déverminage de fiabilité sous contraintes

Le déverminage de fiabilité sous contraintes s'applique aux matériels conformément à 6.7.7 de la CEI 60300-2. En revanche, il ne s'applique pas aux logiciels.

6.9 Programme de coût du cycle de vie

Il convient que le programme de coût du cycle de vie soit conforme à 6.8 de la CEI 60300-2.

Il convient de considérer le coût du cycle de vie d'un logiciel comme une partie du coût dans le programme de coût du cycle de vie du produit ou du système.

6.10 Planification de l'exploitation et de la logistique de maintenance

6.10.1 Planification de la logistique de maintenance

Il convient que la planification de la logistique de maintenance soit conforme à 6.9.1 de la CEI 60300-2.

Il est conseillé que la planification de la logistique de maintenance des logiciels comporte une évaluation de l'effort de maintenance, une attribution des tâches et responsabilités pour l'autorisation de mise en service planifiée ou non du logiciel, une actualisation ou une modification du logiciel ainsi qu'une maintenance perfective visant à améliorer le logiciel. Il convient que la logistique de maintenance soit intégrée au plan de soutien de l'ensemble du projet. Il est recommandé que le soutien logistique, y compris la répartition des ressources, le déploiement des installations et du matériel, la documentation et la formation du personnel, soient planifiés.

6.10.2 Installation

Il convient que l'installation soit conforme à 6.9.2 de la CEI 60300-2.

L'installation du logiciel consiste à suivre le plan du projet concernant la livraison et l'installation du logiciel dans le système et comprend la réception et la mise en service finales du logiciel pour qu'il puisse fonctionner au sein du système dans son environnement réel. Dans ce cadre, le code et les banques de données du logiciel sont initialisés, les essais habituels réalisés et terminés comme spécifié dans les spécifications du contrat et du plan d'essai. Il convient de documenter les étapes et résultats de l'installation pour faciliter les actions de suivi.

6.10.3 Service de soutien

Il convient que le service de soutien soit conforme à 6.9.3 de la CEI 60300-2.

Le service de soutien concernant les logiciels comprend les activités continues de soutien relatives à la maintenance et à l'amélioration du logiciel dans le système en exploitation. Le service de soutien peut être effectué par le fournisseur du logiciel, par un tiers ou par les utilisateurs du logiciel en tenant compte des instructions appropriées. Dans chaque cas, une formation du personnel de soutien est essentielle pour garantir le service de soutien.

6.10.4 Ingénierie de soutien

Il convient que l'ingénierie de soutien soit conforme à 6.9.4 de la CEI 60300-2.

L'ingénierie de soutien comprend le travail d'ingénierie, la connaissance et les qualifications exigées par le personnel de soutien pour satisfaire aux exigences lors du service de soutien.

6.8.7 Reliability stress screening

Reliability stress screening in accordance with 6.7.7 of IEC 60300-2 is applicable to hardware products. It is not applicable to software.

6.9 Life cycle cost programme

Life cycle cost programme should be in accordance with 6.8 of IEC 60300-2.

Software life cycle cost should be treated as a cost element in the product or system life cycle cost programme.

6.10 Operation and maintenance support planning

6.10.1 Maintenance support planning

Maintenance support planning should be in accordance with 6.9.1 of IEC 60300-2.

Planning for software maintenance support includes estimation of maintenance effort and assignment of tasks and responsibilities for scheduled and unscheduled software release, software update or modification, and perfective maintenance for software enhancement. Maintenance support plans should form part of the overall project support plan. Logistics support, including resource allocation, facility and equipment deployment, documentation, and training of personnel, should also be considered as part of the planning effort.

6.10.2 Installation

Installation should be in accordance with 6.9.2 of IEC 60300-2.

Software installation is the execution of the project plan for software delivery and installation in the system on site, final acceptance and commissioning of the software to operate in the system in its actual environment. In this respect, the software code and databases are initialized, the test routines are executed and terminated as specified in the contract specification and test plan. The installation events and results should be documented to facilitate follow-up actions.

6.10.3 Support services

Support services should be in accordance with 6.9.3 of IEC 60300-2.

Support services for software include the continuous support activities for the maintenance and upgrade of the software in an operating system. Support services may be performed by the supplier of the software, or they may be performed by contracting to a third party, or they may be performed by the users of the software with proper instructions. In all cases, training of support personnel is crucial to the success of the support services.

6.10.4 Support engineering

Support engineering should be in accordance with 6.9.4 of IEC 60300-2.

Support engineering is the engineering effort, the knowledge and skills required by the support personnel to fulfil the requirement in performing the support services.

6.10.5 Approvisionnement des rechanges

L'approvisionnement des rechanges concerne le matériel et ne s'applique pas au logiciel.

6.11 Améliorations et modifications

6.11.1 Programmes d'amélioration

Il convient que les programmes d'amélioration soient conformes à 6.10.1 de la CEI 60300-2.

L'amélioration du logiciel est liée à la maintenance du logiciel. Les programmes d'amélioration du logiciel pourraient consister, par exemple, à améliorer les caractéristiques du logiciel afin de fournir une plus grande capacité de mémoire, ou à simplifier les procédures administratives et les procédures de documentation afin d'obtenir une exploitation plus rentable.

Dans chaque cas, il est recommandé de conserver la trace des informations relatives aux étapes successives pour fournir des indications sur les tendances du processus d'amélioration.

Il convient de prendre en compte la maintenance corrective et perfective pour améliorer ou faire évoluer le logiciel dans le processus de maintenance identifié à l'annexe C.

6.11.2 Gestion des modifications

Il convient que la gestion des modifications soit conforme à 6.10.2 de la CEI 60300-2.

Il est conseillé que la gestion des modifications du logiciel soit conforme au processus de gestion de configuration du logiciel qui comprend les procédures administratives et techniques appropriées. Elle consiste à identifier, à enregistrer et à rapporter les états des modifications pour assurer leur complétude, leur cohérence et leur exactitude pour la maintenance de la qualité et de l'efficacité en continu du service.

Il convient de considérer la gestion des modifications résultant de la maintenance corrective et perfective du logiciel dans le processus de gestion de configuration identifié à l'annexe C.

6.12 Retour d'expérience

6.12.1 Acquisition des données

Il convient que l'acquisition des données soit conforme à 6.11.1 de la CEI 60300-2.

Il est recommandé que l'acquisition des données soit principalement une collecte de données du produit ou du système, principalement à partir de l'exploitation sur le terrain et du retour d'expérience des utilisateurs. Il convient que les résultats tirés des essais et de la vérification et de la validation du logiciel fassent partie des données. Il convient que le système de collecte des données soit simple et approprié afin de fournir les données essentielles nécessaires à l'analyse de la disponibilité. Dans le meilleur des cas, il convient de mettre de côté les données brutes liées aux défaillances du matériel, aux défauts du logiciel et à des erreurs de procédures afin de les analyser ultérieurement. En conséquence, il convient de considérer la conception de la méthode d'acquisition des données et du système de collecte des données.

6.10.5 Spares provisioning

Spares provisioning is a hardware activity and not applicable to software.

6.11 Improvements and modifications

6.11.1 Improvement programmes

Improvement programmes should be in accordance with 6.10.1 of IEC 60300-2.

Improvement of software is related to software maintenance. Examples of improvement programmes related to software could be an upgrade of the software features to provide more storage capabilities, or the simplification of administrative or documentation procedures to achieve cost-effective operations.

In all cases, event data should be maintained to provide indications of improvement trends.

Corrective and perfective maintenance for software improvement should be considered in the maintenance process as identified in annex C.

6.11.2 Modification control

Modification control should be in accordance with 6.10.2 of IEC 60300-2.

Software modification control should conform to the established software configuration management process where appropriate administrative and technical procedures are applied. This is to identify, record, and report on the status of the modification to ensure its completeness, consistency and correctness for maintenance of continuous service quality and effectiveness.

Modification control resulting from corrective and perfective maintenance for software should be considered in the configuration management process as identified in annex C.

6.12 Experience feedback

6.12.1 Data acquisition

Data acquisition should be in accordance with 6.11.1 of IEC 60300-2.

Data acquisition should focus on data collection of product or system performance, primarily from field operation and experience feedback from users. Results from test cases and software verification and validation should be included as part of the data. The data collection system should be simple and adequate to provide the essential data necessary for analysis of availability performance. In an ideal situation, the raw data associated with hardware failures, software faults, and procedural errors should be easily segregated for further analysis. Hence, the design of the data acquisition procedure and data collection system should be considered.

6.12.2 Analyse des données

Il convient que l'analyse des données soit conforme à 6.11.2 de la CEI 60300-2.

L'analyse des données est essentielle pour indiquer la disponibilité et pour identifier les anomalies afin de mettre en place, selon le cas, une action corrective et préventive. L'analyse des informations sur le logiciel tirées des jeux d'essais, des résultats des essais, des données de performances sur le terrain ou de toute autre source pertinente pourrait fournir des informations importantes telles que la surveillance de la croissance de fiabilité, le moment idéal de mise à disposition du logiciel et les problèmes systémiques pour l'analyse des causes. Il convient de préciser clairement l'objet de l'analyse des données dans le plan du projet. Il convient d'interpréter et de revoir toutes les données analysées pour les décisions de gestion et les actions ultérieures afin d'améliorer sans cesse le logiciel.

7 Adaptation des programmes de sûreté de fonctionnement

L'adaptation est un processus qui consiste à adapter ou moduler les exigences de manière à ce qu'elles respectent l'objectif spécifique du projet.

Il convient que l'ensemble des critères indiqués à l'article 5 de la CEI 60300-2 constituent la base du processus d'adaptation.

Les activités générales du processus d'adaptation sont les suivantes:

– identifier l'environnement du projet qui reflète la politique de l'organisation et les infrastructures;
– analyser les exigences du contrat, la criticité et l'impact des produits à livrer, l'aptitude et les ressources disponibles pour la réalisation du projet;
– sélectionner les éléments et les tâches de sûreté de fonctionnement applicables au projet;
– documenter les raisons des choix en formalisant les décisions d'adaptation faisant partie du plan.

Les indications suivantes fournissent des informations supplémentaires afin de faciliter l'adaptation des programmes de sûreté de fonctionnement applicables aux systèmes ou aux produits comportant un logiciel.

L'annexe B donne les lignes directrices concernant la sélection principale (première étape) et la réalisation d'éléments du programme de sûreté de fonctionnement pour les produits spécifiques comportant un logiciel. La méthode de sélection prend en compte:

– les phases du cycle de vie du produit applicables au projet;
– les éléments de sûreté de fonctionnement significatifs pour ces étapes du cycle de vie du produit;
– les processus de cycle de vie du logiciel associé liés aux éléments de sûreté de fonctionnement identifiés;
– les activités du processus du logiciel spécifique sélectionnées pour la réalisation du projet.

L'association entre les processus du cycle de vie du logiciel et les phases du cycle de vie du produit est présentée à l'annexe D. Cette annexe met aussi en correspondance les éléments et les tâches du programme de sûreté de fonctionnement applicables aux produits comportant un logiciel, indiqués à l'article 6 de la CEI 60300-2, avec les éléments de gestion de sûreté de fonctionnement et les éléments génériques du projet selon la CEI 60300-1/ISO 9000-4. Les processus respectifs du cycle de vie du logiciel sont utilisés pour identifier les activités spécifiques du logiciel en vue d'une adaptation et d'une amélioration ultérieures (deuxième étape) lors de la mise en oeuvre du projet.

6.12.2 Data analysis

Data analysis should be in accordance with 6.11.2 of IEC 60300-2.

Data analysis is essential to provide availability performance trends and to identify anomalies for initiation of corrective or preventive action, as appropriate. Analysis of software data derived from test cases, test results, field performance data or from other relevant sources could provide valuable insights and information such as monitoring reliability growth, maturity indication for software release and systemic problems for root cause analysis. The objective for data analysis should be clearly stated in the project plan. All analysed data should be interpreted and reviewed for management decisions and follow-up actions to effect the continuous improvement process.

7 Tailoring of dependability programmes

Tailoring is a process for matching the requirements to meet a specific project objective.

The criteria set forth in clause 5 of IEC 60300-2 should form the basis for the tailoring process.

The general tailoring process activities include:

– identification of the project environment reflecting the organizational policy and infrastructure;
– analysis of the contract requirements, criticality and impact of the deliverables, capability and resources available for project implementation;
– selection of applicable dependability elements and tasks relevant to the project;
– documenting the rationale in formalising the tailoring decisions as part of the project plan.

The following provides additional information to facilitate tailoring of dependability programmes applicable to systems or products containing software.

Annex B provides general guidance for the principal (first level) selection and implementation of relevant dependability programme elements for specific products containing software. The selection process takes into consideration:

– the product life cycle phases applicable to the project;
– the dependability elements relevant to that part of the product life cycle phases;
– the associated software life cycle processes related to the dependability elements identified;
– the specific software process activities selected for project implementation.

The association of the software life cycle processes with the product life cycle phases is presented in annex D. This annex also cross-references the respective dependability programme elements and tasks applicable to products containing software according to clause 6 of IEC 60300-2, and the dependability management elements and project generic elements according to IEC 60300-1/ISO 9000-4. The respective software life cycle processes are used to identify software specific activities for further tailoring and refinement (second level) in project task implementation.

L'annexe C fournit des notes explicatives sur les processus du cycle de vie du logiciel. Chaque processus applicable comprend un ensemble d'activités spécifiques concernant la réalisation du projet de logiciel. La méthode d'adaptation a besoin d'être améliorée ultérieurement (deuxième étape) au niveau de la sélection et de la mise en correspondance des activités du processus du logiciel avec le programme de sûreté de fonctionnement lorsque des composants logiciels sont impliqués. Des informations détaillées sur les activités spécifiques du processus du logiciel sont incluses dans l'ISO/CEI 12207 [2].

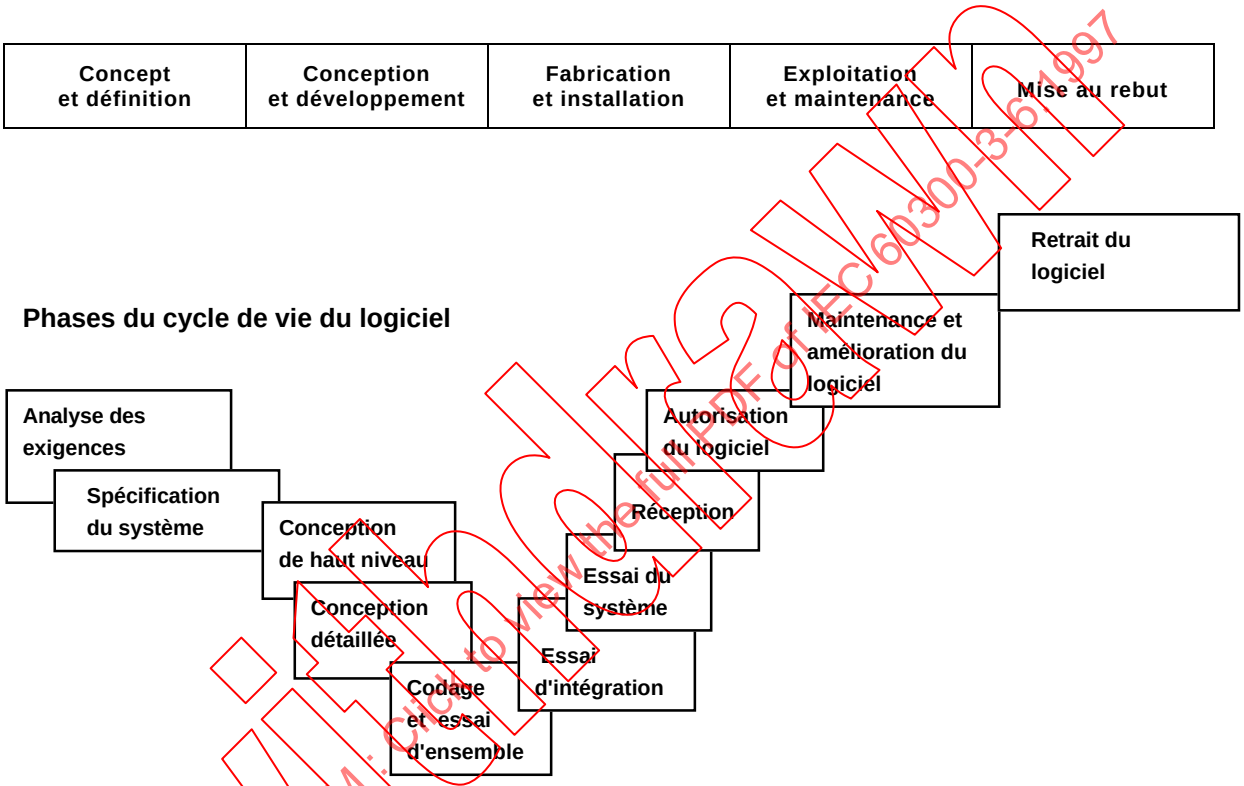
Lorsqu'une adaptation est effectuée sur un programme de sûreté de fonctionnement, il convient de réaliser une étude des coûts afin de respecter les objectifs du projet spécifique. Il est recommandé de rationaliser le travail de sûreté de fonctionnement sélectionné pour la réalisation du programme afin de s'assurer que les activités choisies présentent une valeur ajoutée.

Annex C provides explanatory notes on the software life cycle processes. Each applicable process contains a set of specific activities for software project implementation. The tailoring process requires further refinement (second level) in the selection and mapping of the appropriate software process activities into the dependability programme when software components are involved. Detailed information on specific software process activities is included in ISO/IEC 12207 [2].

Cost consideration should be given when tailoring a dependability programme to meet specific project objectives. Dependability effort selected for programme implementation should be rationalized to ensure that the selected activities add value.

Annexe A
(informative)

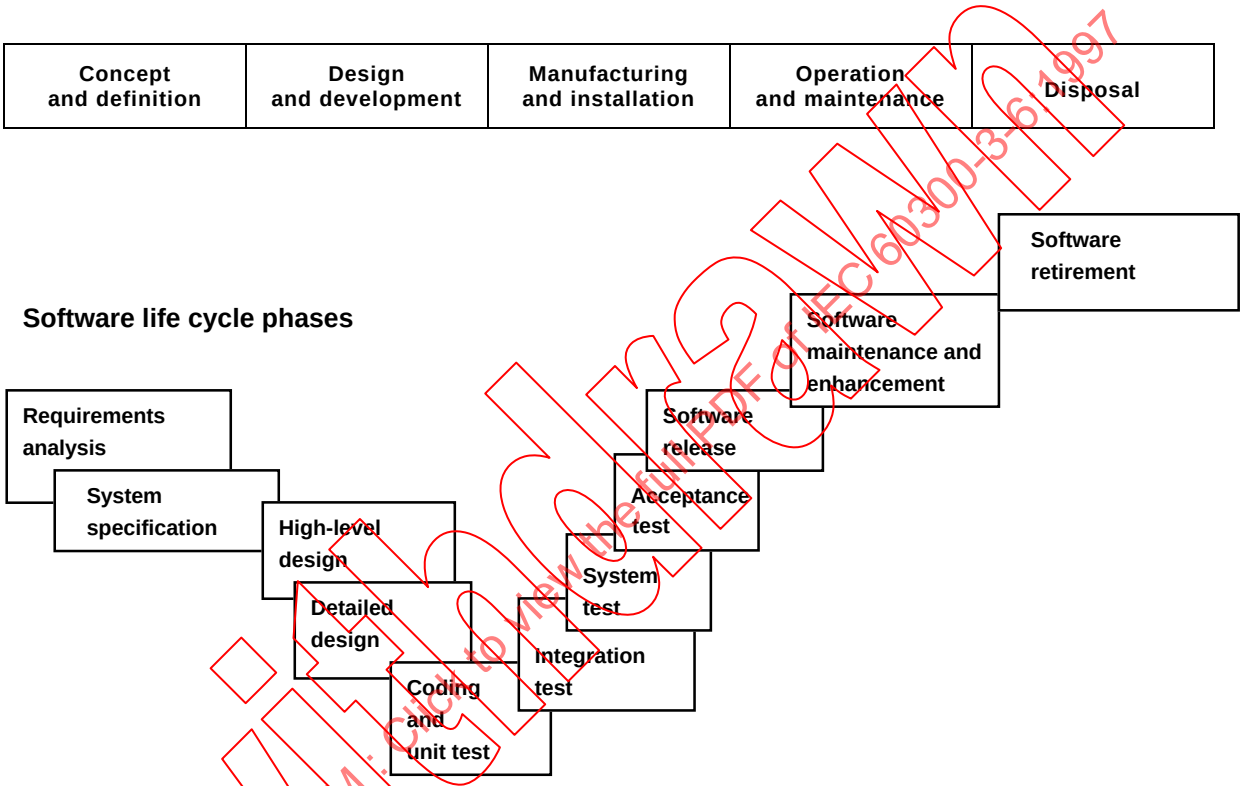
Relation typique entre les phases du cycle de vie du produit et les phases du cycle de vie du logiciel



NOTE – Ce schéma montre les relations entre les étapes du cycle de vie d'un logiciel et les étapes du cycle de vie d'un produit classique à partir d'une représentation séquentielle typique applicable aux programmes de sûreté de fonctionnement.

Annex A
(informative)

Typical relationship of product life cycle phases and software life cycle phases



NOTE – This chart shows the relationship of the software life cycle phases and the conventional product life cycle phases in a typical time-phase representation applicable to dependability programmes.

Annexe B
(informative)

Sélection des éléments du programme de sûreté de fonctionnement

Phases du cycle de vie du produit	Phases du cycle de vie du logiciel	Eléments du programme de sûreté de fonctionnement (CEI 60300-2)										
		6.1 Planification et gestion	6.2 Revue de contrat et liaison	6.3 Exigences de sûreté de fonctionnement	6.4 Ingénierie	6.5 Produits fournis par des tiers	6.6 Analyse, prévision et revues de conception	6.7 Vérification, validation, et essai	6.8 Programme de coût du cycle de vie	6.9 Planification de l'exploitation et de la logistique de maintenance	6.10 Améliorations et modifications	6.11 Retour d'expérience
Concept et définition	Analyse des exigences	X	X	X	X	X	X	X	X	X	X	
Conception et développement	Spécification du système	X	X	X	X	X	X	X	X	X	X	
	Conception de haut niveau	X	X	X	X	X	X	X	X	X	X	
	Conception détaillée	X	X	X	X	X	X	X	X	X	X	X
	Codage et essai d'ensemble	X	X	X	X	X	X	X	X	X	X	X
Fabrication et installation	Essai d'intégration	X	X		X	X	X	X	X	X	X	X
	Essai du système	X	X		X	X	X	X	X	X	X	X
	Réception	X	X		X	X	X	X	X	X	X	X
Exploitation et maintenance	Autorisation du logiciel	X	X		X	X	X	X	X	X	X	X
	Maintenance et amélioration du logiciel	X	X		X	X	X	X	X	X	X	X
Mise au rebut	Retrait du logiciel	X								X		

NOTE – Ce tableau représente une association des phases du cycle de vie du logiciel avec les éléments du programme de sûreté de fonctionnement applicables, à des fins d'identification de premier niveau des tâches du processus d'adaptation décrit à l'article 7.

Annex B
(informative)

Selection of dependability programme elements

Product life cycle phases	Software life cycle phases	Dependability programme elements (IEC 60300-2)										
		6.1 Planing and management	6.2 Contract review and liaison	6.3 Dependability requirements	6.4 Engineering	6.5 Externally provided products	6.6 Analysis, prediction and design review	6.7 Verification, validation and test	6.8 Life cycle cost programme	6.9 Operation and maintenance support planning	6.10 Improvement and modifications	6.11 Experiences feedback
Concept and definition	Requirements analysis	X	X	X	X	X	X	X	X	X	X	
Design and development	System specification	X	X	X	X	X	X	X	X	X	X	
	High-level design	X	X	X	X	X	X	X	X	X	X	
	Detailed design	X	X	X	X	X	X	X	X	X	X	X
	Coding and unit test	X	X	X	X	X	X	X	X	X	X	X
Manufacturing and installation	Integration test	X	X		X	X	X	X	X	X	X	X
	System test	X	X		X	X	X	X	X	X	X	X
	Acceptance test	X	X		X	X	X	X	X	X	X	X
Operation and maintenance	Software release	X	X		X	X	X	X	X	X	X	X
	Software maintenance and enhancement	X	X		X	X	X	X	X	X	X	X
Disposal	Software retirement	X								X		

NOTE – This chart presents a typical association of the software life cycle phases and the applicable dependability programme elements for first level task identification in the tailoring process as described in clause 7.

Annexe C
(informative)

Processus du cycle de vie du logiciel

C.1 Généralités

Les processus du cycle de vie d'un logiciel sont décrits dans l'ISO/CEI 12207 [2]. Ces processus s'appliquent à l'acquisition de systèmes, de logiciels et de services. Ils s'appliquent à la fourniture, au développement, à l'exploitation et à la maintenance des produits logiciels. Un produit logiciel est un ensemble de programmes informatiques, de procédures ainsi que de documentation et d'informations associées. Un processus est un ensemble d'activités liées entre elles qui transforment les entrées en sorties dans le cadre de la mise en oeuvre et de l'exécution du processus.

Les indications suivantes présentent une vue générale des processus du cycle de vie du logiciel afin de faciliter leur association avec les éléments et les tâches du programme de sûreté de fonctionnement.

Les processus du cycle de vie du logiciel sont classés de la manière suivante:

a) les processus principaux du cycle de vie;

b) les processus de soutien du cycle de vie;

c) les processus organisationnels du cycle de vie.

C.2 Processus principaux du cycle de vie

Les processus principaux du cycle de vie sont ceux que prennent en charge les principales personnes responsables du développement, de l'exploitation ou de la maintenance des produits logiciels. Il s'agit de l'acquéreur, du fournisseur, du développeur, de l'opérateur et de la personne chargée de la maintenance des produits logiciels. Il existe cinq processus principaux du cycle de vie.

a) **Le processus d'acquisition** définit les activités de l'acquéreur, c'est-à-dire de l'organisation qui acquiert un système, un produit ou un service logiciel. Les activités du processus d'acquisition comprennent:
   - l'initiation de l'acquisition;
   - la demande d'élaboration d'une proposition ou d'un appel d'offre;
   - la préparation et l'actualisation du contrat;
   - le contrôle du fournisseur;
   - la réception et l'achèvement du contrat.

b) **Le processus de livraison** définit les activités du fournisseur, c'est-à-dire de l'organisation qui fournit le système, le produit logiciel ou le service logiciel à l'acquéreur. Les activités du processus de livraison comprennent:
   - l'initiation du processus de livraison;
   - la préparation de la réponse à l'appel d'offre;
   - le contrat;
   - la planification;
   - l'exécution et le contrôle;
   - la revue et l'évaluation;
   - la livraison et l'achèvement du contrat.

Annex C
(informative)

Software life cycle processes

C.1 General

Software life cycle processes are described in ISO/IEC 12207 [2]. These processes apply to the acquisition of systems, software products and services. They are applicable to the supply, development, operation and maintenance of software products. A software product is a set of computer programs, procedures, and associated documentation and data. A process is a set of interrelated activities, which transform inputs into outputs, in the context of process implementation and throughput.

The following provides an overview of the software life cycle processes to facilitate their association with dependability programme elements and tasks.

The software life cycle processes are grouped into:

a) primary life cycle processes;
b) supporting life cycle processes;
c) organizational life cycle processes.

C.2 Primary life cycle processes

The primary life cycle processes are those processes that serve the primary parties who initiate or perform in the development, operation or maintenance of software products. These primary parties are the acquirer, the supplier, the developer, the operator, and the maintainer of software products. There are five primary life cycle processes.

a) **The acquisition process** defines the activities of the acquirer, that is the organization that acquires a system, software product or software service. The acquisition process activities include:
   – initiation of the acquisition;
   – request for proposal or tender preparation;
   – contract preparation and update;
   – supplier monitoring;
   – acceptance and completion.

b) **The supply process** defines the activities of the supplier, that is the organization that provides the system, software product or software service to the acquirer. The supply process activities include:
   – initiation of the supply process;
   – preparation of response;
   – contract;
   – planning;
   – execution and control;
   – review and evaluation;
   – delivery and completion.

c) **Le processus de développement** définit les activités du développeur, c'est-à-dire de l'organisation qui définit et développe le produit logiciel. Les activités du processus de développement comprennent:

- la mise en oeuvre du processus de développement;
- l'analyse des exigences du système;
- la conception de l'architecture du système;
- l'analyse des exigences du logiciel;
- la conception de l'architecture du logiciel;
- la conception détaillée du logiciel;
- le codage et les essais du logiciel;
- l'intégration du système;
- les essais de qualification du logiciel;
- l'intégration du logiciel;
- les essais de qualification du système;
- l'installation du logiciel;
- le soutien de la réception du logiciel.

d) **Le processus d'exploitation** définit les activités de l'opérateur, c'est-à-dire de l'organisation qui fournit le service de l'exploitation d'un système logiciel dans son environnement réel pour ses utilisateurs. Les activités du processus d'exploitation comprennent:

- la mise en oeuvre du processus d'exploitation;
- les essais en exploitation;
- l'exploitation du système;
- l'aide fournie à l'utilisateur.

e) **Le processus de maintenance** définit les activités de l'opérateur de maintenance, c'est-à-dire de l'organisation qui fournit le service de maintenance pour le logiciel; il s'agit de la gestion des modifications apportées au logiciel pour qu'il reste actuel et en bon état de marche. Les activités du processus de maintenance comprennent:

- la mise en oeuvre du processus de maintenance;
- l'analyse des modifications et des problèmes;
- la mise en oeuvre des modifications;
- la réception/la revue de maintenance;
- le passage de l'ancien au nouvel environnement d'exploitation;
- le retrait du logiciel.

C.3 Processus de soutien du cycle de vie

Les processus de soutien du cycle de vie s'expliquent de la manière suivante: chaque processus soutient un autre processus et devient ainsi partie intégrante de ce dernier. Un processus de soutien est utilisé et exécuté, si nécessaire, par un autre processus. Un processus de soutien a une application et un but spécifiques. Il contribue au succès et à la qualité de l'ensemble du projet de logiciel. Il existe huit processus de soutien du cycle de vie.

a) **Le processus de documentation** définit les activités d'enregistrement des informations produites par un processus du cycle de vie. Les activités du processus de documentation comprennent:

c) **The development process** defines the activities of the developer, that is the organization that defines and develops the software product. The development process activities include:

- development process implementation;
- system requirements analysis;
- system architectural design;
- software requirements analysis;
- software architectural design;
- software detailed design;
- software coding and testing;
- software integration;
- software qualification testing;
- system integration;
- system qualification testing;
- software installation;
- software acceptance support.

d) **The operation process** defines the activities of the operator, that is the organization that provides the service of operating a software system in its live environment for its users. The operation process activities include:

- operation process implementation;
- operational testing;
- system operation;
- user support.

e) **The maintenance process** defines the activities of the maintainer, that is the organization that provides the service of maintaining the software product; it consists in managing modifications to the software product to keep it current and in operational fitness. The maintenance process activities include:

- maintenance process implementation;
- problem and modification analysis;
- modification implementation;
- maintenance review/acceptance;
- migration from old operating environment to new operating environment;
- software retirement.

C.3 Supporting life cycle processes

The supporting life cycle processes are those processes that, individually, each supports another process and forms an integral part with the process it supports. A supporting process is employed and executed, as needed, by another process. A supporting process has its specific application and purpose. It contributes to the success and quality of the overall software project. There are eight supporting life cycle processes.

a) **The documentation process** defines the activities for recording the information produced by a life cycle process. The documentation process activities include:

- la mise en oeuvre du processus de documentation;
- la conception et le développement des documents;
- la production des documents;
- la maintenance des documents.

b) **Le processus de gestion de la configuration** définit les activités de gestion de configuration concernant l'administration et le contrôle des logiciels de base dans un système au niveau de leur mise à disposition ou des modifications. Les activités de gestion de configuration comprennent:

- la mise en oeuvre du processus de gestion de la configuration;
- l'identification de la configuration;
- le contrôle de la configuration;
- la prise en compte de l'état de la configuration;
- l'évaluation de la configuration;
- la gestion de la mise à disposition et la livraison.

c) **Le processus d'assurance de la qualité** définit les activités visant à assurer de manière objective la conformité des produits logiciels et des processus avec les exigences spécifiées et le respect des plans établis. Les activités du processus d'assurance de la qualité comprennent:

- la mise en oeuvre du processus d'assurance de la qualité;
- l'assurance produit;
- l'assurance processus;
- l'assurance des systèmes qualité.

d) **Le processus de vérification** définit les activités concernant la vérification des produits logiciels du projet. Les activités du processus de vérification comprennent:

- la mise en oeuvre du processus de vérification;
- la vérification de l'exécution du contrat, du processus, des exigences, de la conception, du code, de l'intégration et de la documentation comme prévu dans le projet.

e) **Le processus de validation** définit les activités concernant la validation des produits logiciels du projet. Les activités du processus de validation comprennent:

- la mise en place du processus de validation;
- la validation en effectuant des essais et une analyse des résultats des essais.

f) **Le processus de revue en commun** définit les activités concernant l'évaluation de l'état et des produits d'une activité. Les activités du processus de revue en commun comprennent:

- la mise en oeuvre du processus de revue en commun;
- les revues de gestion du projet;
- les revues techniques.

g) **Le processus d'audit** définit les activités visant à déterminer la conformité avec les exigences, les plans et le contrat. Les activités du processus d'audit comprennent:

- la mise en oeuvre du processus d'audit;
- la mise en oeuvre de l'audit.

- documentation process implementation;
- design and development of the documents;
- production of the documents;
- maintenance of the documents.

b) **The configuration management process** defines the configuration management activities on the administration and control of the baseline software items in a system for their release or modification. The configuration management process activities include:

- configuration management process implementation;
- configuration identification;
- configuration control;
- configuration status accounting;
- configuration evaluation;
- release management and delivery.

c) **The quality assurance process** defines the activities for objectively assuring that the software products and processes are in conformance with their specified requirements and adhere to their established plans. The quality assurance process activities include:

- quality assurance process implementation;
- product assurance;
- process assurance;
- assurance of quality systems.

d) **The verification process** defines the activities for verifying the software products of the software project. The verification process activities include:

- implementation of the verification process;
- conducting contract verification, process verification, requirements verification, design verification, code verification, integration verification, and documentation verification as appropriate to the software project.

e) **The validation process** defines the activities for validating the software products of the software project. The validation process activities include:

- implementation of the validation process;
- performing validation by means of test cases and analysis of test results.

f) **The joint review process** defines the activities for evaluating the status and products of an activity. The joint review process activities include:

- implementation of the joint review process;
- project management reviews;
- technical reviews.

g) **The audit process** defines the activities for determining compliance with the requirements, plans and contract. The audit process activities include:

- implementation of the audit process;
- conducting the audit.

h) **Le processus de résolution des problèmes** définit les activités visant à analyser et à résoudre les problèmes détectés lors du développement, de l'exploitation et de la maintenance du logiciel. Les activités du processus de résolution des problèmes comprennent:

- la mise en oeuvre du processus de résolution des problèmes;
- la résolution des problèmes.

C.4 Processus organisationnels du cycle de vie

Les processus organisationnels du cycle de vie sont utilisés par une organisation afin d'établir et de mettre en place une infrastructure sous-jacente composée de processus de cycle de vie associés, centrés sur le client, gérés par la direction, y compris les installations et le personnel pour l'amélioration continue des infrastructures et des processus. Il existe quatre processus de cycle de vie organisationnels.

a) **Le processus de gestion** définit les activités de base de gestion, y compris la gestion du projet, pendant le processus du cycle de vie. Les activités du processus de gestion comprennent:

- l'initiation et la définition du domaine d'application;
- la planification;
- l'exécution et le contrôle;
- la revue et l'évaluation;
- la clôture.

b) **Le processus d'infrastructure** définit les activités de base concernant la mise en place de la structure de base d'un processus du cycle de vie. Les activités du processus d'infrastructure comprennent:

- la mise en place du processus d'infrastructure;
- la mise en place de l'infrastructure;
- la maintenance de l'infrastructure.

c) **Le processus d'amélioration** définit les activités de base qu'une organisation réalise afin d'établir, de mesurer, de contrôler et d'améliorer son processus de cycle de vie. Les activités du processus d'amélioration comprennent:

- la mise en oeuvre du processus d'amélioration;
- l'évaluation du processus;
- l'amélioration du processus.

d) **Le processus de formation** définit les activités visant à fournir un personnel formé de manière correcte. Les activités du processus de formation comprennent:

- la mise en oeuvre du processus;
- le développement des éléments nécessaires à la formation;
- la mise en place d'un plan de formation.

h) **The problem resolution process** defines a process for analyzing and resolving the problems discovered during software development, operation, and maintenance. The problem resolution process activities include:

- implementation of the problem resolution process;
- problem resolution.

C.4 Organizational life cycle processes

The organizational life cycle processes are employed by an organization to establish and implement an underlying infrastructure, which is made up of associated life cycle processes, customer focus, management leadership, facilities and personnel, for continuous improvement of the infrastructure and processes. There are four organizational life cycle processes.

a) **The management process** defines the basic activities of the management, including project management, during a life cycle process. The management process activities include:

- initiation and scope definition;
- planning;
- execution and control;
- review and evaluation;
- closure.

b) **The infrastructure process** defines the basic activities for establishing the underlying structure of a life cycle process. The infrastructure process activities include:

- implementation of the infrastructure process;
- establishment of the infrastructure;
- maintenance of the infrastructure.

c) **The improvement process** defines the basic activities that an organization performs for establishing, measuring, controlling and improving its life cycle process. The improvement process activities include:

- improvement process establishment;
- process assessment;
- process improvement.

d) **The training process** defines the activities for providing adequately trained personnel. The training process activities include:

- process implementation;
- training material development;
- training plan implementation.

Annexe D
(informative)

Association des processus du cycle de vie du logiciel avec les phases du cycle de vie du produit

Processus du cycle de vie du logiciel (ISO/CEI 12207)	Elément et tâches du programme de sûreté de fonctionnement (CEI 60300-2)	Phases du cycle de vie du produit (CEI 60300-1)				
		Concept et définiton	Conception et développement	Fabrication et installation	Exploitation et maintenance	Mise au rebut
Processus principaux du cycle de vie						
5.1 Processus d'acquisition	6.2 Revue de contrat et liaison	xxxxxxxxxx				
	6.5 Produits fournis par des tiers		xxxxxxxxx			
5.2 Processus de livraison	6.2 Revue de contrat et liaison	xxxxxxxxxx				
5.3 Processus de développement	6.3 Exigences de sûreté de fonctionnement	xxxxxxxxxxxx				
	6.4 Ingénierie	xxxxxxxxx	xxxxxxxxxxxx			
	6.6 Analyse, prévision et revues de conception	xxxxxxxxx	xxxxxxxxxx			
	6.7 Vérification, validation et essai			xxxxxxxxxxxxxxxxx		
5.5 Processus d'exploitation	6.8 Programme de coût du cycle de vie	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
	6.9 Planification de l'exploitation et de la logistique de maintenance				xxxxxxxxxxxxxxx	
5.5 Processus de maintenance	6.9 Planification de l'exploitation et de la logistique de maintenance				xxxxxxxxxxxxxxx	
	6.10 Améliorations et modifications				xxxxxxxxxxxxxxx	
	6.11 Retour d'expérience				xxxxxxxxxxxxxxx	
Processus de soutien du cycle de vie						
6.1 Processus de documentation	6.11 Retour d'expérience				xxxxxxxxxxxxxxx	
	Enregistrement des données de sûreté de fonctionnement[1)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
6.2 Processus de gestion de la configuration	6.1.4 Gestion de configuration	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
6.3 Processus d'assurance de la qualité	Systèmes qualité[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
6.4 Processus de vérification	6.7 Vérification, validation et essai			xxxxxxxxxxxxxxxxx		
6.5 Processus de validation	6.7 Vérification, validation et essai			xxxxxxxxxxxxxxxxx		
6.6 Processus de revue en commun	6.6.8 Revues de conception formalisées		xxxxxxxxxx			
6.7 Processus d'audit	Systèmes qualité[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
6.8 Processus de résolution des problèmes	Revue de gestion[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
	Revue du programme de sûreté de fonctionnement[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
Processus organisationnels du cycle de vie						
7.1 Processus de gestion	Politique de sûreté de fonctionnement[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
	Mise en oeuvre de la gestion de la sûreté de fonctionnement[1)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
7.2 Processus d'infrastructure	Organisation[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
7.3 Processus d'amélioration	Systèmes qualité[2)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx
7.4 Processus de formation	Mise en oeuvre de la gestion de la sûreté de fonctionnement[1)]	xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxx	xxxxxxxxx

[1)] Eléments génériques du projet selon la CEI 60300-1/ISO 9000-4.
[2)] Eléments de gestion de la sûreté de fonctionnement selon la CEI 60300-1/ISO 9000-4.

NOTE – Ce tableau représente l'association des processus de cycle de vie du logiciel avec les éléments du programme de sûreté de fonctionnement applicables, à des fins d'identification de premier niveau des tâches du processus d'adaptation décrit à l'article 7. Le séquencement indiqué dans la colonne «Phases du cycle de vie du produit» CEI 60300-1 illustre des exemples typiques en relation avec les éléments et les tâches du programme de sûreté de fonctionnement conformément à la CEI 60300-2.