NORME INTERNATIONALE

INTERNATIONAL STANDARD

CEI
IEC
62138

Première édition
First edition
2004-01

**Centrales nucléaires –
Instrumentation et contrôle-commande importants pour la sûreté –
Aspects logiciels des systèmes informatisés réalisant des fonctions de catégorie B ou C**

**Nuclear power plants –
Instrumentation and control important for safety –
Software aspects for computer-based systems performing category B or C functions**

Numéro de référence
Reference number
CEI/IEC 62138:2004

Numérotation des publications

Depuis le 1er janvier 1997, les publications de la CEI sont numérotées à partir de 60000. Ainsi, la CEI 34-1 devient la CEI 60034-1.

Editions 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.

Informations supplémentaires sur les publications de la CEI

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 cette publication, y compris sa validité, sont disponibles dans le Catalogue des publications de la CEI (voir ci-dessous) en plus des nouvelles éditions, amendements et corrigenda. Des informations sur les sujets à l'étude et l'avancement des travaux entrepris par le comité d'études qui a élaboré cette publication, ainsi que la liste des publications parues, sont également disponibles par l'intermédiaire de:

- **Site web de la CEI (www.iec.ch)**
- **Catalogue des publications de la CEI**

  Le catalogue en ligne sur le site web de la CEI (www.iec.ch/searchpub) vous permet de faire des recherches en utilisant de nombreux critères, comprenant des recherches textuelles, par comité d'études ou date de publication. Des informations en ligne sont également disponibles sur les nouvelles publications, les publications remplacées ou retirées, ainsi que sur les corrigenda.
- **IEC Just Published**

  Ce résumé des dernières publications parues (www.iec.ch/online_news/justpub) est aussi disponible par courrier électronique. Veuillez prendre contact avec le Service client (voir ci-dessous) pour plus d'informations.
- **Service clients**

  Si vous avez des questions au sujet de cette publication ou avez besoin de renseignements supplémentaires, prenez contact avec le Service clients:

  Email: custserv@iec.ch
  Tél: +41 22 919 02 11
  Fax: +41 22 919 03 00

Publication numbering

As from 1 January 1997 all IEC publications are issued with a designation in the 60000 series. For example, IEC 34-1 is now referred to as IEC 60034-1.

Consolidated editions

The IEC is now publishing consolidated versions of its publications. 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.

Further information on IEC publications

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 this publication, including its validity, is available in the IEC Catalogue of publications (see below) in addition to new editions, amendments and corrigenda. Information on the subjects under consideration and work in progress undertaken by the technical committee which has prepared this publication, as well as the list of publications issued, is also available from the following:

- **IEC Web Site (www.iec.ch)**
- **Catalogue of IEC publications**

  The on-line catalogue on the IEC web site (www.iec.ch/searchpub) enables you to search by a variety of criteria including text searches, technical committees and date of publication. On-line information is also available on recently issued publications, withdrawn and replaced publications, as well as corrigenda.
- **IEC Just Published**

  This summary of recently issued publications (www.iec.ch/online_news/justpub) is also available by email. Please contact the Customer Service Centre (see below) for further information.
- **Customer Service Centre**

  If you have any questions regarding this publication or need further assistance, please contact the Customer Service Centre:

  Email: custserv@iec.ch
  Tel: +41 22 919 02 11
  Fax: +41 22 919 03 00

NORME INTERNATIONALE

INTERNATIONAL STANDARD

CEI
IEC
62138

Première édition
First edition
2004-01

Centrales nucléaires – Instrumentation et contrôle-commande importants pour la sûreté – Aspects logiciels des systèmes informatisés réalisant des fonctions de catégorie B ou C

Nuclear power plants – Instrumentation and control important for safety – Software aspects for computer-based systems performing category B or C functions

International Electrotechnical Commission, 3, rue de Varembé, PO Box 131, CH-1211 Geneva 20, Switzerland
Telephone: +41 22 919 02 11 Telefax: +41 22 919 03 00 E-mail: inmail@iec.ch Web: www.iec.ch

Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

CODE PRIX
PRICE CODE X

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

SOMMAIRE

CONTENTS

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

CENTRALES NUCLÉAIRES – INSTRUMENTATION ET CONTRÔLE-COMMANDE IMPORTANTS POUR LA SÛRETÉ – ASPECTS LOGICIELS DES SYSTÈMES INFORMATISÉS RÉALISANT DES FONCTIONS DE CATÉGORIE B OU C

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (CEI) 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, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). 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 de la CEI intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) La CEI n'a prévu aucune procédure de marquage valant indication d'approbation et n'engage pas sa responsabilité pour les équipements déclarés conformes à une de ses Publications.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI 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 62138 a été établie par le sous-comité 45A: Instrumentation et contrôle-commande des installations nucléaires, du comité d'études 45 de la CEI: Instrumentation nucléaire.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
45A/507/FDIS	45A/521/RVD

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

Cette publication a été rédigée selon la Partie 2 des Directives ISO/CEI.

INTERNATIONAL ELECTROTECHNICAL COMMISSION

NUCLEAR POWER PLANTS – INSTRUMENTATION AND CONTROL IMPORTANT FOR SAFETY – SOFTWARE ASPECTS FOR COMPUTER-BASED SYSTEMS PERFORMING CATEGORY B OR C FUNCTIONS

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of 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, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). 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. 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 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 IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62138 has been prepared by subcommittee 45A: Instrumentation and control of nuclear facilities, of IEC technical committee 45: Nuclear instrumentation.

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

FDIS	Report on voting
45A/507/FDIS	45A/521/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.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant 2009. A cette date, la publication sera

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

The committee has decided that the contents of this publication will remain unchanged until 2009. At this date, the publication will be:

- reconfirmed;
- withdrawn;
- replaced by a revised edition, or
- amended.

INTRODUCTION

Structure de la collection de normes du SC 45A – Relations avec les documents de la CEI, de l'AIEA et de l'ISO

Le point d'entrée de la collection de normes produite par le SC 45A est la CEI 61513. Cette norme traite des exigences relatives aux systèmes et équipements d'instrumentation et de contrôle-commande (systèmes d'I&C) utilisés pour accomplir les fonctions importantes pour la sûreté des centrales nucléaires, et structure la collection de normes du SC 45A.

La CEI 61513 fait directement référence aux autres normes du SC 45A traitant de sujets génériques tels que la catégorisation des fonctions et le classement des systèmes, la qualification, la séparation des systèmes, les aspects logiciels et les aspects matériels pour les systèmes informatisés, la conception des salles de commande et le multiplexage. Ces normes directement référencées forment avec la CEI 61513 un ensemble documentaire cohérent.

Les normes du SC 45A qui ne sont pas référencées directement par la CEI 61513 sont relatives à des matériels particuliers, à des méthodes, à des techniques ou à des activités spécifiques. Généralement, ces documents de bas niveau font référence aux documents de plus haut niveau décrits précédemment pour les activités génériques, et peuvent être utilisés de façon isolée.

La CEI 61513 a adopté une présentation similaire à celle de la CEI 61508, avec un cycle de vie et de sûreté global, un cycle de vie et de sûreté des systèmes, et une interprétation des exigences générales des parties 1, 2 et 4 de la CEI 61508 pour le secteur nucléaire. La conformité à la CEI 61513 facilite la compatibilité avec les exigences de la CEI 61508 telles qu'elles ont été interprétées dans l'industrie nucléaire.

La CEI 61513 fait référence aux normes ISO ainsi qu'au document AIEA 50-C-QA (qui a depuis été remplacé par le document AIEA 50-C/SG-Q) pour ce qui concerne l'assurance qualité.

Les normes produites par le SC 45A sont élaborées de façon à être en accord avec les principes de sûreté fondamentaux du Code AIEA sur la sûreté des centrales nucléaires, ainsi qu'avec les guides de sûreté de l'AIEA, en particulier le guide NS-R-1 "Safety of Nuclear Power Plants: Design – Requirements" et le guide NS-G-1.3 "Instrumentation and Control Systems Important to Safety in Nuclear Power Plants – Safety Guide". La terminologie et les définitions utilisées dans les normes produites par le SC 45A sont conformes à celles utilisées par l'AIEA.

INTRODUCTION

Structure of the SC 45A standard series – Relationships with other IEC, IAEA and ISO documents

The entry point of the SC 45A standard series is IEC 61513. This standard deals with general requirements for instrumentation and control systems and equipment (I&C systems) that are used to perform functions important to safety in nuclear power plants (NPPs), and structures the SC45A standard series.

IEC 61513 refers directly to other SC 45A standards for general topics related to categorization of functions and classification of systems, qualification, separation of systems, software aspects of computer-based systems, hardware aspect of computer-based systems, control rooms design and multiplexing. The standards referenced directly have to be considered together with IEC 61513 as a consistent document set.

The other SC 45A standards not directly referenced by IEC 61513 are standards related to particular equipment, technical methods or specific activities. Usually, those low level documents, which refer to the documents of the higher levels previously described for the general topics, can be used on their own.

IEC 61513 has adopted a presentation format similar to basic safety publication IEC 61508, with an overall safety lifecycle frame and a system safety lifecycle frame, and provides an interpretation of the general requirements of IEC 61508, parts 1, 2 and 4, for the nuclear application sector. Compliance with IEC 61513 will facilitate consistency with the requirements of IEC 61508 as they have been interpreted for the nuclear industry. In that frame, IEC 60880 and IEC 62138 correspond to IEC 61508, part 3 for the nuclear application sector.

IEC 61513 refers to ISO as well as to IAEA 50-C-QA (now replaced by IAEA 50-C/SG-Q) for topics related to quality assurance.

The SC 45A standards series implements consistently and in detail the principles and basic safety aspects given in the IAEA Code on the safety of nuclear power plants and in the IAEA safety series, in particular the Requirements NS-R-1, “Safety of Nuclear Power Plants: Design” and the Safety Guide NS-G-1.3, “Instrumentation and Control Systems Important to Safety in Nuclear Power Plants”. The terminology and definitions used by the SC 45A standards are consistent with that used by the IAEA.

CENTRALES NUCLÉAIRES – INSTRUMENTATION ET CONTRÔLE-COMMANDE IMPORTANTS POUR LA SÛRETÉ – ASPECTS LOGICIELS DES SYSTÈMES INFORMATISÉS RÉALISANT DES FONCTIONS DE CATÉGORIE B OU C

1 Domaine d'application

Cette Norme internationale énonce des exigences sur les logiciels des systèmes d'instrumentation et de contrôle-commande (I&C) informatisés réalisant des fonctions de sûreté de catégorie B ou C, selon la définition donnée par la CEI 61226. Elle est complémentaire à la CEI 60880 et à la CEI 60880-2, qui énoncent des exigences sur le logiciel des systèmes d'I&C informatisés réalisant des fonctions de sûreté de catégorie A.

Elle est également cohérente et complémentaire à la CEI 61513. Les activités de nature essentiellement système (comme l'intégration, la validation et l'installation sur site) n'y sont pas traitées exhaustivement: les exigences qui ne sont pas spécifiques au logiciel sont à chercher dans la CEI 61513.

La CEI 61513 définit ainsi la classe des systèmes d'I&C importants pour la sûreté:

- les systèmes d'I&C de classe 1 sont principalement prévus pour réaliser des fonctions de catégorie A, mais peuvent aussi réaliser des fonctions de catégorie B et/ou C, ainsi que des fonctions non classées;
- les systèmes d'I&C de classe 2 sont principalement prévus pour réaliser des fonctions de catégorie B, mais peuvent aussi réaliser des fonctions de catégorie C, ainsi que des fonctions non classées;
- les systèmes d'I&C de classe 3 sont principalement prévus pour réaliser des fonctions de catégorie C, mais peuvent aussi réaliser des fonctions non classées.

Un système d'I&C classé de sûreté pouvant réaliser des fonctions de catégories différentes, ainsi que des fonctions non classées, les exigences de cette Norme sont directement attachées à la catégorie de sûreté des fonctions supportées, mais à la classe de sûreté du système.

Cette Norme prend en compte les pratiques de développement actuellement mises en oeuvre pour les logiciels de systèmes d'I&C, et en particulier:

- l'utilisation de logiciels, d'équipements et de familles d'équipements pré-développés, mais pas nécessairement selon les normes de l'industrie nucléaire;
- l'utilisation de «boîtes noires» contenant du logiciel;
- l'utilisation de langages orientés application.

Cette Norme n'est pas conçue comme un guide général de génie logiciel. Elle énonce les exigences que les logiciels des systèmes d'I&C de classe 2 et 3 doivent satisfaire afin d'atteindre les objectifs de sûreté nucléaire du système.

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

NUCLEAR POWER PLANTS – INSTRUMENTATION AND CONTROL IMPORTANT FOR SAFETY – SOFTWARE ASPECTS FOR COMPUTER-BASED SYSTEMS PERFORMING CATEGORY B OR C FUNCTIONS

1 Scope

This International Standard provides requirements for the software of computer-based I&C systems performing functions of safety category B or C as defined by IEC 61226. It complements IEC 60880 and IEC 60880-2, which provide requirements for the software of computer-based I&C systems performing functions of safety category A.

It is also consistent with, and complementary to, IEC 61513. Activities that are mainly system level activities (for example, integration, validation and installation) are not addressed exhaustively by this standard: requirements that are not specific to software are deferred to IEC 61513.

IEC 61513 defines the safety classes of I&C systems important to safety as follows:

- I&C systems of safety class 1 are basically intended to perform functions of safety category A, but may also perform functions of safety category B and/or C, and non safety-classified functions;
- I&C systems of safety class 2 are basically intended to perform functions of safety category B, but may also perform functions of safety category C, and non safety-classified functions;
- I&C systems of safety class 3 are basically intended to perform functions of safety category C, but may also perform non safety-classified functions.

Since a given safety-classified I&C system may perform functions of different safety categories and even non safety-classified functions, the requirements of this standard are attached to the safety class of the I&C system.

This standard takes into account the current practices for the development of software for I&C systems, in particular:

- the use of pre-developed software, equipment and equipment families that were not necessarily designed to nuclear industry sector standards;
- the use of dedicated “black-box” devices with embedded software;
- the use of application-oriented languages.

This standard is not intended to be used as a general-purpose software engineering guide. It provides requirements that the software of I&C systems of safety classes 2 or 3 must meet to achieve system nuclear safety objectives.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

CEI 61226, *Centrales nucléaires – Systèmes d'instrumentation et de contrôle-commande importants pour la sûreté – Classification*

CEI 61513:2001, *Centrales nucléaires – Instrumentation et contrôle commande des systèmes importants pour la sûreté – Prescriptions générales pour les systèmes*

3 Termes, définitions et abréviations

Pour les besoins du présent document, les termes, définitions et abréviation suivants s'appliquent.

3.1
animation
processus par lequel le comportement défini par une spécification est visualisé avec ses valeurs effectives dérivées des équations de comportement et des valeurs d'entrée

(CEI 60880-2)

3.2
fonction d'application
fonction d'un système d'I&C qui accomplit une tâche relative au processus sous contrôle plutôt qu'au fonctionnement du système lui-même

(CEI 61513)

3.3
langage orienté application
langage informatique spécifiquement conçu pour un certain type d'application et pour être utilisé par les spécialistes de ce type d'application

NOTE 1 Les familles d'équipements offrent en général des langages orientés application de façon à faciliter l'adaptation des équipements à des besoins particuliers.

NOTE 2 Les langages orientés application peuvent être utilisés pour la spécification d'exigences fonctionnelles que doit satisfaire un système d'I&C, ou pour spécifier ou concevoir le logiciel d'application. Ils peuvent être basés sur du texte, des diagrammes, ou une combinaison des deux.

NOTE 3 Exemples: les langages à blocs fonctionnels, les langages définis par la CEI 61131-3.

NOTE 4 Voir aussi Langage généraliste.

3.4
logiciel d'application
partie du logiciel d'un système d'I&C qui exécute des fonctions d'application

(CEI 61513)

NOTE Voir aussi Logiciel système, Logiciel système opérationnel.

3.5
catégorie d'une fonction d'I&C
une des trois affectations possibles (A, B ou C) des fonctions d'I&C, résultant de l'évaluation de l'importance pour la sûreté des fonctions à exécuter. Une affectation «non classée» peut être délivrée si la fonction n'est pas importante pour la sûreté

(CEI 61513)

NOTE Voir aussi Classe d'un système d'I&C.

IEC 61226, *Nuclear power plants – Instrumentation and control systems important for safety – Classification*

IEC 61513:2001, *Nuclear power plants – Instrumentation and control for systems important to safety – General requirements for systems*

3 Terms, definitions and abbreviations

For the purposes of this document, the following terms, definitions and abbreviation apply.

3.1
animation
process by which the behaviour defined by a specification is displayed with actual values derived from the stated behaviour expressions and from some input values

(IEC 60880-2)

3.2
application function
function of an I&C system that performs a task related to the process being controlled rather than to the functioning of the system itself

(IEC 61513)

3.3
application-oriented language
computer language specifically designed to address a certain type of application and to be used by persons who are specialists of this type of application

NOTE 1 Equipment families usually feature application-oriented languages so as to provide easy to use capability for adjusting the equipment to specific requirements.

NOTE 2 Application-oriented languages may be used to specify the functional requirements of an I&C system, and/or to specify or design application software. They may be based on texts, on graphics, or on both.

NOTE 3 Examples: function block diagram languages, languages defined by IEC 61131-3.

NOTE 4 See also General-purpose language.

3.4
application software
part of the software of an I&C system that implements the application functions

(IEC 61513)

NOTE See also System software, Operational system software.

3.5
category of an I&C function
one of three possible safety assignments (A, B, C) of I&C functions resulting from considerations of the importance to safety of the functions to be performed. An unclassified assignment may be made if the function is not significant to safety

(IEC 61513)

NOTE See also Class of an I&C system.

3.6
classe d'un système d'I&C
une des trois affectations possibles (1, 2 ou 3) des systèmes d'I&C importants pour la sûreté, résultant de la nécessité pour ces systèmes d'exécuter des fonctions d'I&C d'importances pour la sûreté différentes. Une affectation «non classée» est délivrée si le système d'I&C n'exécute pas de fonction importante pour la sûreté

(CEI 61513)

NOTE Voir aussi Catégorie d'une fonction d'I&C.

3.7
complexité
degré de difficulté à comprendre ou vérifier la conception, la réalisation ou le comportement d'un système ou d'un composant

(CEI 61513)

3.8
gestion de configuration
discipline appliquant des mesures techniques et administratives pour identifier et documenter les caractéristiques fonctionnelles et physiques d'un élément de configuration, maîtriser les modifications apportées à ces caractéristiques, enregistrer et signaler les évolutions de statut et vérifier la conformité aux exigences spécifiées

(CEI 61513)

3.9
documentation de conception
document ou ensemble de documents décrivant l'organisation et le fonctionnement d'une entité, et qui est à la base de la réalisation et de l'intégration

3.10
documentation pour la sûreté
document ou ensemble de documents spécifiant comment un produit peut être utilisé en sûreté pour des applications importantes pour la sûreté

3.11
famille d'équipements
ensemble de composants matériels et logiciels pouvant travailler de manière complémentaire dans une variété d'architectures (configurations). Le développement des configurations spécifiques à la centrale et du logiciel d'application associé peut être réalisé à l'aide d'outils logiciels. Une famille d'équipements fournit normalement un certain nombre de fonctionnalités standards (bibliothèque de fonctions d'application) qui peuvent être combinées pour générer un logiciel d'application spécifique

(CEI 61513)

NOTE 1 Une famille d'équipements peut être un produit provenant d'un fabricant ou un ensemble de produits interconnectés et adaptés par un fournisseur.

NOTE 2 Le terme «plate-forme» est parfois utilisé comme synonyme de «famille d'équipements».

3.12
erreur
différence entre une valeur ou condition calculée, observée ou mesurée et la valeur ou condition réelle, spécifiée ou théorique

(CEI 61513)

NOTE Vois aussi Erreur humaine, Défaut, Défaillance.

3.6
class of an I&C system
one of three possible assignments (1, 2, 3) of I&C systems important to safety resulting from consideration of their requirement to implement I&C functions of differing importance to safety. An unclassified assignment is made if the I&C system does not implement functions important to safety

(IEC 61513)

NOTE See also Category of an I&C function.

3.7
complexity
degree to which a system or component has a design, implementation or behaviour that is difficult to understand and verify

(IEC 61513)

3.8
configuration management
discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control modifications to those characteristics, record and report changes in status, and verify compliance with specified requirements

(IEC 61513)

3.9
design specification
document or set of documents that describe the organisation and functioning of an item, and that are used as a basis for the implementation and the integration of the item

3.10
documentation for safety
document or set of documents that specifies how a product can be safely used for applications important to safety

3.11
equipment family
set of hardware and software components that may work co-operatively in one or more defined architectures (configurations). The development of plant specific configurations and of the related application software may be supported by software tools. An equipment family usually provides a number of standard functionalities (application functions library) that may be combined to generate specific application software

(IEC 61513)

NOTE 1 An equipment family may be a product of a defined manufacturer or a set of products interconnected and adapted by a supplier.

NOTE 2 The term “Equipment platform” is sometime used as a synonym of “Equipment family”.

3.12
error
discrepancy between a computed, observed or measured value or condition, and the true, specified or theoretical value or condition

(IEC 61513)

NOTE See also Mistake, Fault, Failure.

3.13
code exécutable
partie du logiciel faisant partie du système cible

NOTE Le code exécutable comprend normalement les instructions devant être exécutées par le matériel du système cible ainsi que les données associées.

3.14
défaillance
déviation du service délivré par rapport au service attendu

(CEI 61513)

NOTE Voir aussi Erreur humaine, Défaut, Erreur.

3.15
défaut
imperfection dans un composant matériel, logiciel ou système

(CEI 61513)

NOTE 1 Les défauts peuvent être répartis en défauts aléatoires et défauts systématiques. Les défauts aléatoires sont causés par la dégradation du matériel et provoquent des défaillances à des instants qui ne peuvent pas être déterminés à l'avance. Les défauts systématiques résultent d'erreurs de conception, et, dans des conditions identiques, provoquent systématiquement les mêmes défaillances (exemple: les défauts logiciels).

NOTE 2 Un défaut (en particulier un défaut de conception) peut rester indétecté jusqu'à ce qu'une situation particulière conduise à un résultat non conforme, c'est-à-dire à une défaillance.

NOTE 3 Voir aussi Erreur humaine, Erreur, Défaillance.

3.16
validation fonctionnelle
certification de l'adéquation des spécifications des fonctions d'application aux exigences fonctionnelles et de performance de la centrale. Elle est complémentaire de la validation du système, qui vérifie la conformité du système à la spécification des fonctions

(CEI 61513)

3.17
langage généraliste
langage informatique conçu pour s'adresser à tout type de besoin

NOTE 1 Le logiciel système d'une famille d'équipements est en général réalisé à l'aide de langages généraliste.

NOTE 2 Exemples: Ada, C, Pascal.

NOTE 3 Voir aussi Langage orienté application.

3.18
intégration
assemblage et vérification progressifs de composants en un système complet

3.19
architecture d'I&C
structure organisant les systèmes d'I&C de la centrale importants pour la sûreté

(CEI 61513)

3.20
erreur humaine
action humaine (ou inaction) conduisant à un résultat indésirable

(CEI 60880-2)

NOTE Voir aussi Défaut, Erreur, Défaillance.

3.13
executable code
software that is included in the target system

NOTE Executable code usually includes instructions to be executed by the hardware of the target system, and associated data.

3.14
failure
deviation of the delivered service from the intended one

(IEC 61513)

NOTE See also Mistake, Fault, Error.

3.15
fault
defect in a hardware, software or system component

(IEC 61513)

NOTE 1 Faults may be subdivided into random faults and systematic faults. Random faults result from hardware degradation and cause failures at unpredictable times. Systematic faults result from design errors (for example, software faults) and, in identical conditions, lead systematically to the same failures.

NOTE 2 A fault (in particular a design fault) may remain undetected in a system until specific conditions are such that the result produced does not conform to the intended function, i.e. a failure occurs.

NOTE 3 See also Mistake, Error, Failure.

3.16
functional validation
verification of the correctness of the application functions specifications versus the plant functional and performance requirements. It is complementary to the system validation that verifies the compliance of the system with the functions specification

(IEC 61513)

3.17
general-purpose language
computer language designed to address all types of usage

NOTE 1 The system software of equipment families is usually implemented using general-purpose languages.

NOTE 2 Examples: Ada, C, Pascal.

NOTE 3 See also Application-oriented language.

3.18
integration
progressive aggregation and verification of components into a complete system

3.19
I&C architecture
organisational structure of the I&C systems of a plant which are important to safety

(IEC 61513)

3.20
mistake
human action (or inaction) that produces an unintended result

(IEC 60880-2)

NOTE See also Fault, Error, Failure.

3.21
mode de fonctionnement
état fonctionnel d'une entité dans lequel elle offre un comportement opérationnel particulier

NOTE Exemples: initialisation, mode normal, modes dégradés en cas d'erreur dans l'entité ou dans son environnement.

3.22
logiciel système opérationnel
partie du logiciel système dont le code exécutable fonctionne sur le processeur cible pendant le fonctionnement du système

(CEI 61513)

NOTE 1 Exemples: système d'exploitation, gestionnaires d'entrées/sorties et de communication, gestion des exceptions, gestion des tâches, gestion des interruptions, diagnostics en ligne, gestion des redondances et des modes dégradés, bibliothèques d'application.

NOTE 2 Voir aussi Logiciel d'application, Logiciel système.

3.23
paramètre
donnée gouvernant le comportement du système d'I&C et/ou de son logiciel, et pouvant être modifiée par les opérateurs durant l'exploitation

3.24
logiciel pré-développé
logiciel qui existe déjà et qui peut ou non être un produit commercial

(CEI 61513)

NOTE 1 Dans les logiciels pré-développés, on peut distinguer les logiciels indépendants de tout environnement matériel, des logiciels intégrés à des composants matériels et qui doivent être utilisés en association à ces composants.

NOTE 2 Dans la présente norme, ce terme ne couvre pas les outils logiciels, même lorsqu'ils sont pré-développés.

3.25
programme
document écrit par un être humain et qui est transformé en code exécutable par des moyens automatiques

NOTE Ceci inclut les programmes écrits en langages généralistes. Ceci inclut également les programmes écrits en langages orientés application.

3.26
sécurité
capacité d'un système informatique à donner une confiance suffisante dans le fait que les personnes et systèmes non autorisés ne pourront ni modifier le logiciel et ses données, ni accéder aux fonctions du système, et dans le fait que cela ne sera pas refusé aux personnes et systèmes autorisés

(CEI 61513)

3.27
logiciel
programmes (ensembles ordonnés d'instructions), données, règles et toute documentation associée relatifs au fonctionnement d'un système informatique

(CEI 60880)

3.28
modification du logiciel
changement dans un document ou des documents déjà approuvés conduisant à un changement dans le code exécutable

NOTE Une modification du logiciel peut être effectuée durant le développement initial (par exemple pour éliminer des défauts mis en évidence dans les phases finales du développement) ou après la mise en service du logiciel.

3.21
mode of behaviour
functional state of an item where it provides a specific operational behaviour

NOTE Examples: initialisation mode, normal mode, downgraded modes to be taken in case of error in the item or in its environment.

3.22
operational system software
part of system software the executable code of which runs on the target processor during system operation

(IEC 61513)

NOTE 1 Examples: operating system, input/output and communication drivers, exception handlers, scheduler, interrupt management, on-line diagnostic, redundancy and graceful degradation management, application software libraries.

NOTE 2 See also Application software, System software.

3.23
parameter
data item governing the behaviour of the I&C system and/or of its software, and that may be modified by operators during plant operation

3.24
pre-developed software
software part that already exists and is available as a commercial or proprietary product

(IEC 61513)

NOTE 1 Pre-developed software may be divided into software that has not been specifically developed for a specific hardware environment, and software integrated in hardware components that has to be used in association with this hardware.

NOTE 2 In this standard, this term does not cover software tools, even when they are pre-developed.

3.25
program
document written by a human being that is transformed into executable code by automated tools

NOTE This includes traditional programs written in general-purpose languages. This also includes programs written in application-oriented languages.

3.26
security
capability of a computer-based system to provide adequate confidence that unauthorised persons and systems can neither modify the software and its data nor gain access to the system functions, and yet to ensure that this is not denied to authorised persons and systems

(IEC 61513)

3.27
software
programs (i.e. sets of ordered instructions), data, rules and any associated documentation pertaining to the operation of a computer-based I&C system

(IEC 60880)

3.28
software modification
change in an already agreed document (or documents) leading to an alteration of the executable code

NOTE Software modifications may occur either during initial software development (for example, to remove faults found in later stages of development), or after the software is already in service.

3.29
composant logiciel
une des entités constituant un logiciel. Un composant peut lui même être constitué par d'autres composants logiciels

(CEI 61513)

3.30
développement du logiciel
phase du cycle de vie du logiciel conduisant à la création du logiciel d'un système d'I&C ou d'un produit logiciel. Cette phase couvre toutes les activités depuis la spécification d'exigences jusqu'à la validation et l'installation sur le site

3.31
cycle de vie et de sûreté du logiciel
activités nécéssaires au développement et à l'exploitation du logiciel d'un système d'I&C important pour la sûreté. Elles couvrent la période allant de la spécification des exigences sur le logiciel au retrait de service du logiciel

(CEI 61513)

3.32
analyse statique
processus d'évaluation d'un système ou d'un composant basé sur sa forme, sa structure, son contenu ou sa documentation

(CEI 60880-2)

3.33
logiciel système
partie du logiciel d'un système d'I&C, d'un équipement ou d'une famille d'équipements conçue pour faciliter le développement, l'exploitation et la modification de ces systèmes et des programmes associés

(CEI 61513)

NOTE 1 Le logiciel système est généralement composé de logiciels opérationnels et de logiciels de soutien (outils logiciels).

NOTE 2 Voir aussi Logiciel d'application, Logiciel système opérationnel.

3.34
validation du logiciel
test et évaluation d'un logiciel intégré pour s'assurer de sa conformité aux spécifications de fonctionnalité, de performance et d'interface imposées par les exigences sur le système

3.35
vérification
confirmation par examen et apport d'éléments objectifs que les résultats d'une activité sont conformes aux objectifs et exigences établis pour cette activité

(ISO 12207)

3.36
abréviation
I&C: Instrumentation et Contrôle-Commande

3.29
software component
one of the design entities that make up a software item. It may be subdivided into other software components

(IEC 61513)

3.30
software development
phase of the software lifecycle that leads to the creation of the software of an I&C system or of a software product. It covers all the activities from software requirements specification to validation and installation on site

3.31
software safety lifecycle
necessary activities involved in the development and operation of the software of an I&C system important to safety occurring during a period of time that starts with the software requirements specification and finishes when the software is withdrawn from use

(IEC 61513)

3.32
static analysis
process of evaluating a system or component based on its form, structure, content or documentation

(IEC 60880-2)

3.33
system software
part of the software of an I&C system designed for a specific computer or equipment family to facilitate the development, operation and modification of these items and associated programs

(IEC 61513)

NOTE 1 The system software of equipment families is usually composed of operational system software and of support system software (software tools).

NOTE 2 See also Application software, Operational system software.

3.34
software validation
test and evaluation of integrated software to ensure compliance with the functional, performance and interface specifications imposed by the I&C system requirements

3.35
verification
confirmation by examination and by provision of objective evidence that the results of an activity meet the objectives and requirements defined for this activity (ISO 12207)

3.36
abbreviation
I&C: Instrumentation and Control

4 Concepts et présupposés

Cet article présente les principaux concepts et présupposés relatifs à la nature et au développement du logiciel des systèmes d'I&C des classes de sûreté 2 et 3, et sur lesquels le texte normatif repose.

4.1 Types de logiciels

La Figure 1 illustre la variété des services offerts par les logiciels et les composants logiciels d'un système d'I&C ou d'une architecture d'I&C typiques. Les composants logiciels sont souvent répartis entre logiciel système et logiciel d'application. Le logiciel système est lui-même divisé en logiciel système opérationnel, qui est embarqué dans les systèmes d'I&C classés de sûreté, et en logiciel de soutien (ou outils logiciels) qui est hors-ligne ou embarqué dans des systèmes de service non classés de sûreté. Du logiciel peut aussi être trouvé dans des équipements spécialisés tels que des capteurs et des actionneurs, des équipements de communication et des onduleurs.

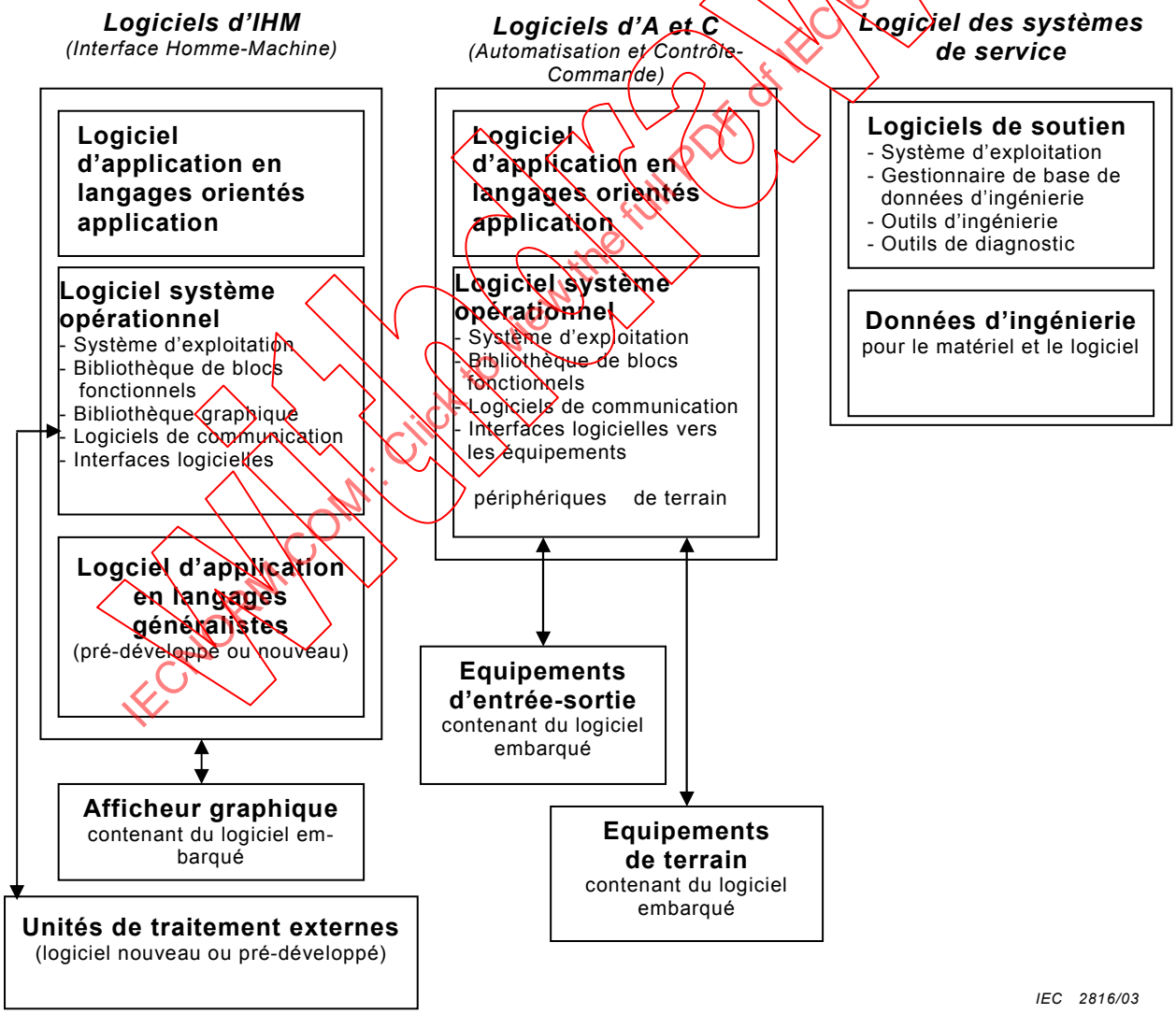

Figure 1 – Composants logiciels typiques d'un système d'I&C informatisé

4 Key concepts and assumptions

This Clause presents some of the key concepts and assumptions about the nature and the development of the software of I&C systems of safety class 2 or 3, upon which the normative text is based.

4.1 Types of software

Figure 1 illustrates the variety of services offered by software and software components in a typical I&C system or I&C architecture. Software components may often be defined as being either system software or application software. System software may also be divided into operational system software, which is embedded in safety classified I&C systems, and support system software (or software tools) which is either off-line or embedded in non-safety classified support systems. Software may also be found in dedicated devices such as sensors and actuators, communication devices and Uninterruptible Power Supplies (UPSs).

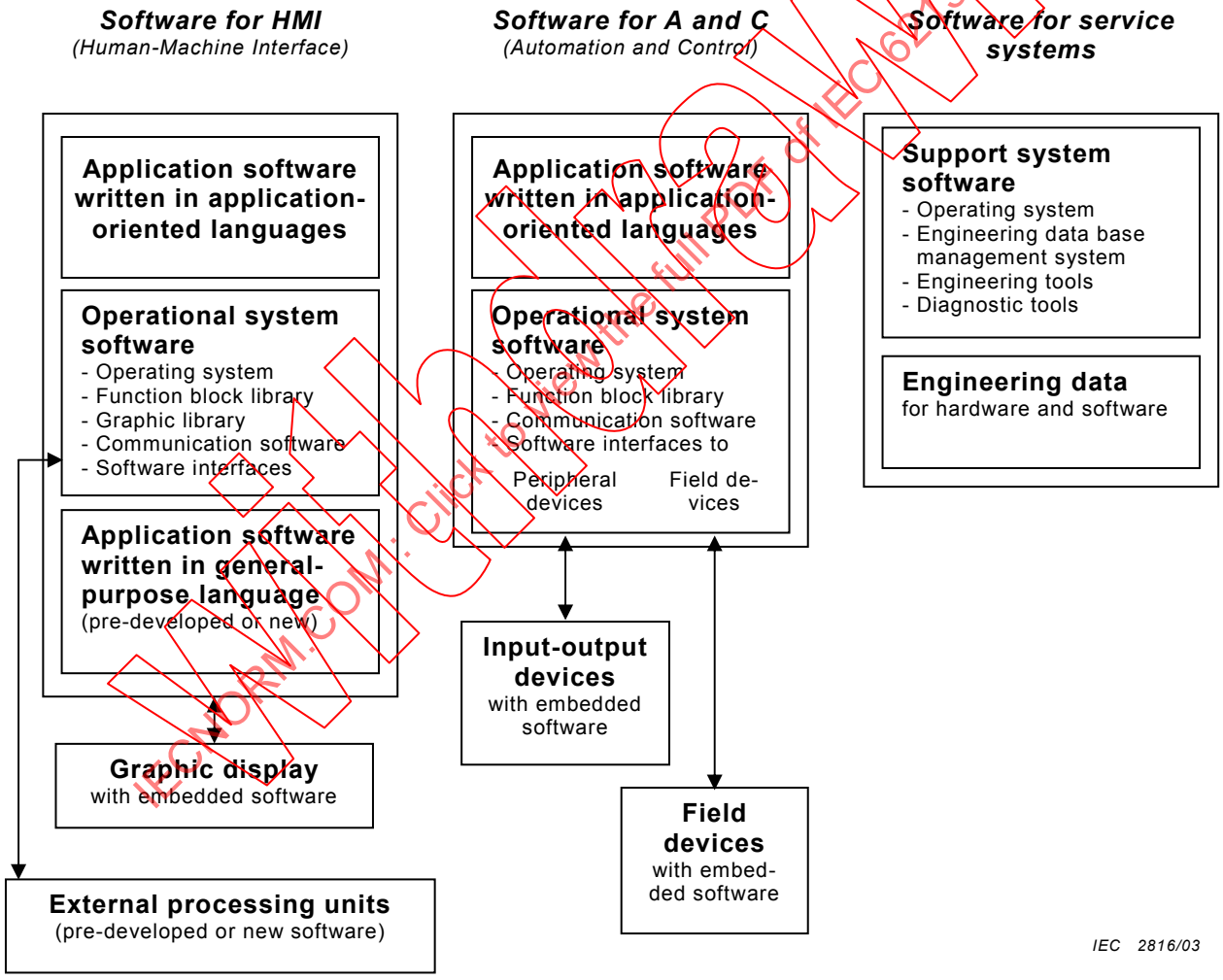

Figure 1 – Typical software parts in computer-based I&C systems

Le logiciel d'un système d'I&C peut aussi être divisé en logiciel pré-développé (offrant le plus souvent des fonctions utiles pour une variété de systèmes d'I&C) et en logiciel nouveau (développé le plus souvent pour les besoins spécifiques d'un système). Le logiciel système est en général pré-développé, et le logiciel d'application est en général nouveau, mais ceci n'est pas une règle absolue. Les exigences de cette Norme qui sont applicables aux logiciels nouveaux peuvent également être appliquées aux logiciels pré-développés. La norme énonce également des exigences de substitution applicables seulement aux logiciels pré-développés et aux équipements pré-développés contenant du logiciel.

De nombreuses familles d'équipements incluent une large panoplie d'outils de développement orientés application permettant aux ingénieurs concevant la centrale ou les systèmes élémentaires de spécifier leurs exigences graphiquement. Des outils peuvent alors traduire automatiquement les programmes graphiques en logiciel d'application exécutable. Lorsque ces outils sont d'une qualité appropriée, il est admis que cette approche permet de réduire les risques de défaut.

4.2 Types de données

La conception de nombreux systèmes fait largement appel à des données de configuration. Une donnée de configuration peut être liée au logiciel système opérationnel ou au logiciel d'application. Les données de configuration liées au logiciel d'application sont en général des données d'ingénierie résultant de la conception de la centrale et sont produites pour l'essentiel par des concepteurs de centrale qui n'ont pas besoin d'une expérience particulière en génie logiciel. Les données de configuration sont divisées en:

- données qui ne peuvent être modifiées par les opérateurs de la centrale, et qui sont soumises aux mêmes exigences que le reste du logiciel;
- paramètres qui peuvent être modifiés par les opérateurs durant l'exploitation de la centrale (par exemple les seuils, les données d'étalonnage) et qui font l'objet d'exigences particulières.

4.3 Cycles de Vie et de Sûreté du Logiciel et du Système

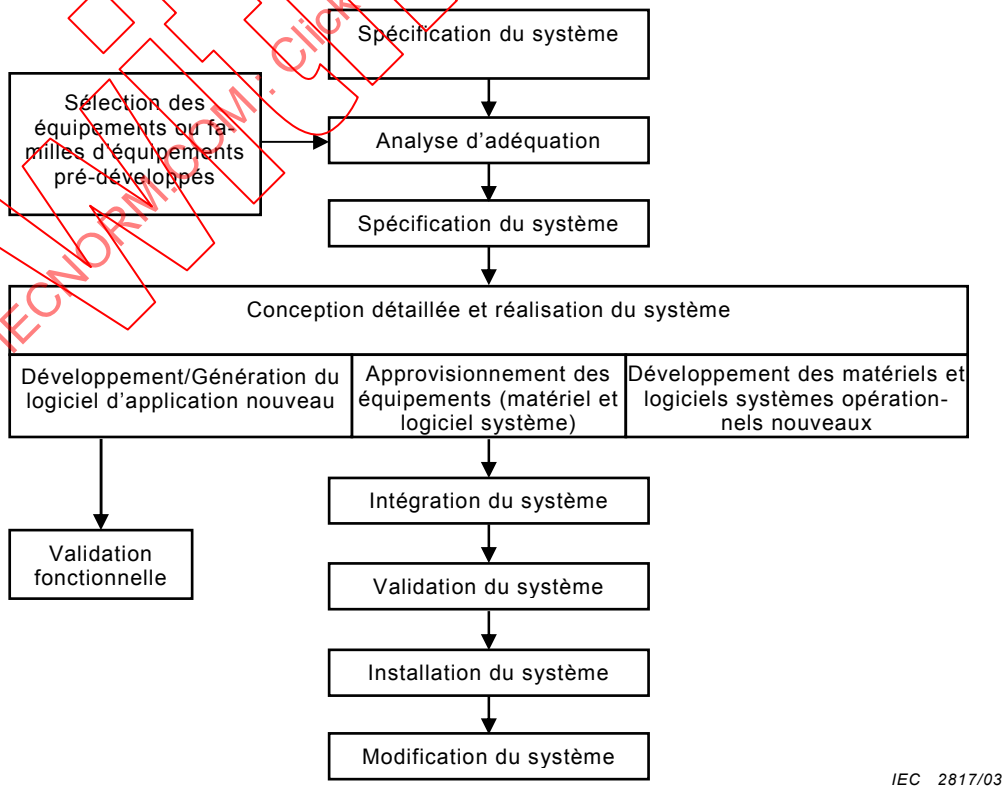

Figure 2 – Activités du Cycle de Vie et de Sûreté du Système (selon la CEI 61513)

The software in an I&C system may also be divided into pre-developed software (which usually provides functions useful to a range of I&C systems) and new software (which is developed to the specific needs of the I&C system). System software is usually pre-developed, and application software is usually new, but this is not an absolute rule. The requirements of this standard that are applicable to new software may also be applied to pre-developed software. The standard also provides alternative requirements that may be applied specifically to pre-developed software or dedicated devices with embedded software.

Many modern equipment families are provided with extensive application-oriented development tools that enable plant or system engineers to specify their requirements using graphical techniques. The tools may automatically translate the graphical programs into executable application software. When these tools are of adequate quality, this approach is considered to reduce the risk of faults.

4.2 Types of data

Many system designs make extensive use of configuration data. Configuration data may be associated with operational system software or with application software. Configuration data associated with application software consists mainly of plant engineering data resulting from the design of the plant, and is often prepared by plant designers who are not required to have software skills. Configuration data may be divided into:

- data items which are not intended to be modified on-line by plant operators, and which are submitted to the same requirements as apply to the rest of the software;
- parameters, i.e., data items which may be modified by operators during plant operation (for example, alarm limits, set points, data required to calibrate instrumentation) and which need specific requirements.

4.3 Software and System Safety Lifecycles

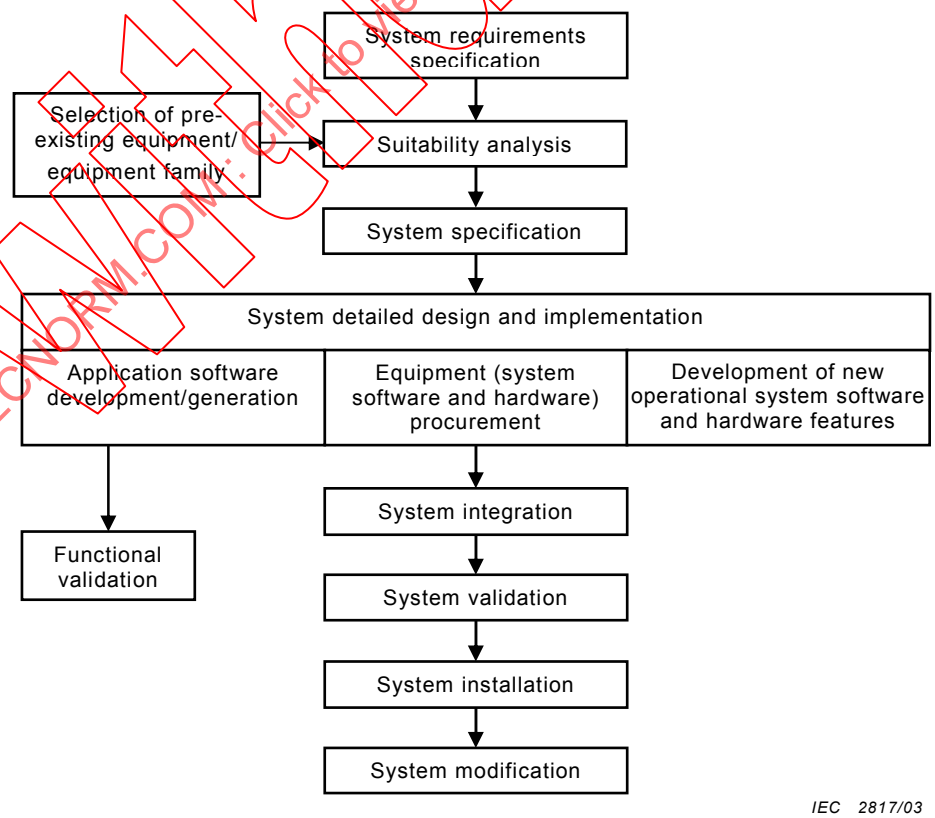

Figure 2 – Activities of the System Safety Lifecycle (as defined by IEC 61513)

Le logiciel contribue en général fortement aux fonctions réalisées par le système d'I&C. Il peut aussi contribuer à des fonctions ajoutées par la conception du système (initialisation et surveillance du matériel, communication entre sous-systèmes et synchronisation par exemple). Le Cycle de Vie et de Sûreté du Logiciel est donc, dans la plupart des cas, fortement intégré au Cycle de Vie et de Sûreté du Système. En particulier, la spécification du logiciel est une partie, ou est déduite directement de la spécification et de la conception du système.

Et bien que la vérification des composants logiciels nouveaux fasse clairement partie du Cycle de Vie et de Sûreté du Logiciel, il n'y a souvent pas de frontière nette entre l'intégration du logiciel et l'intégration du système. Par conséquent, dans cette Norme, l'intégration du logiciel est considérée comme faisant partie de l'intégration du système. De la même façon, dans cette Norme, la validation du logiciel est considérée comme faisant partie de la validation du système.

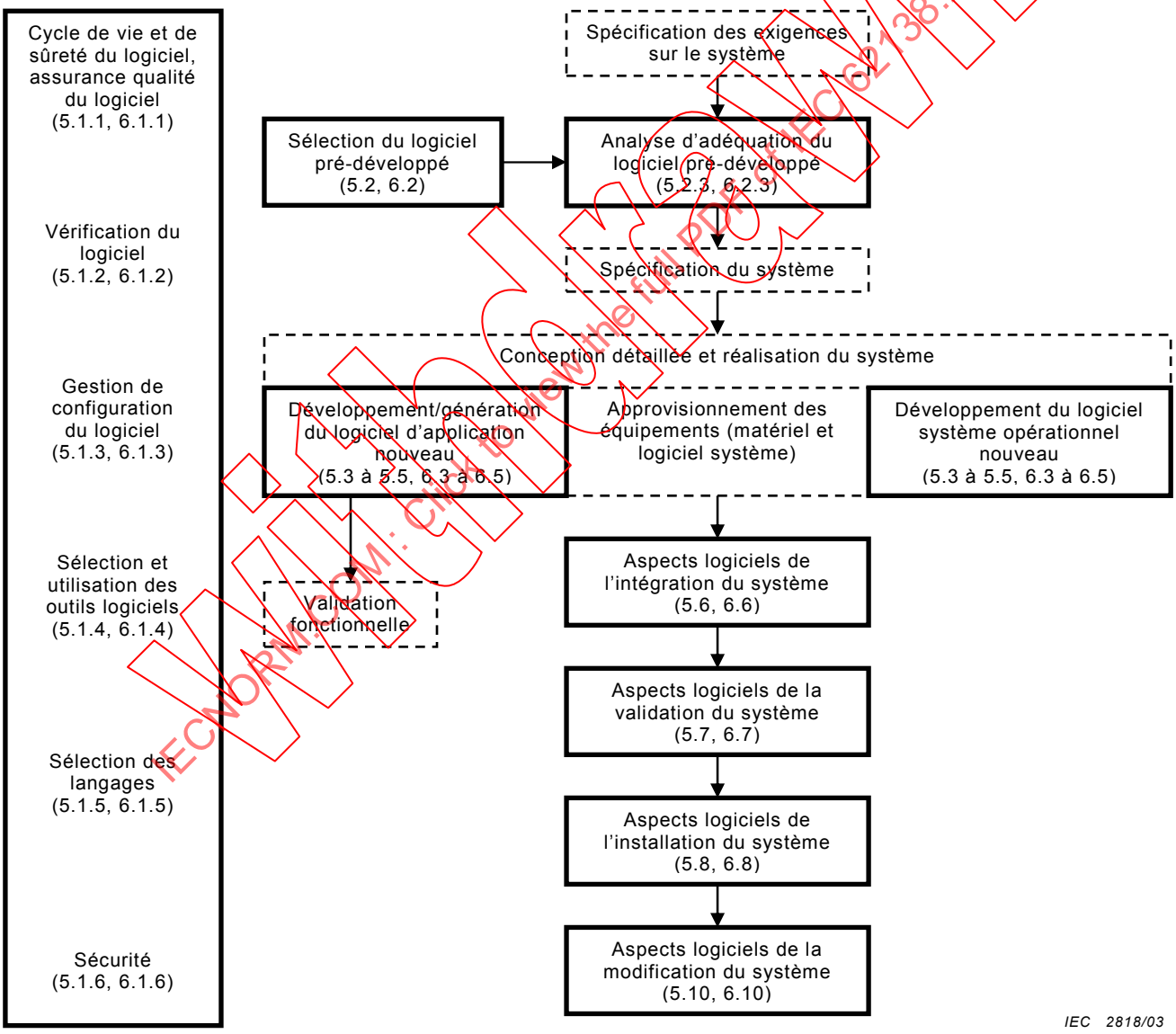

Figure 3 – Activités logicielles dans le Cycle de Vie et de Sûreté du Système

(les cases en traits fins pointillés représentent des activités de niveau système non traitées dans la présente Norme)

Software usually contributes strongly to the functions performed by the I&C system. It may also support additional functions introduced by system design (for example, initialisation and surveillance of hardware, communication between, and synchronisation of, sub-systems). Thus, the Software Safety Lifecycle is in most cases strongly integrated with the System Safety Lifecycle. In particular, the software requirements specification is a part of, or is derived directly from, system specification and system design.

And though the verification of new software components is definitely a part of the Software Safety Lifecycle, there is often no separate and well-identified boundary between software integration and system integration. Therefore, in this standard, software integration is considered to be a part of system integration. Software validation too is considered a part of system validation.

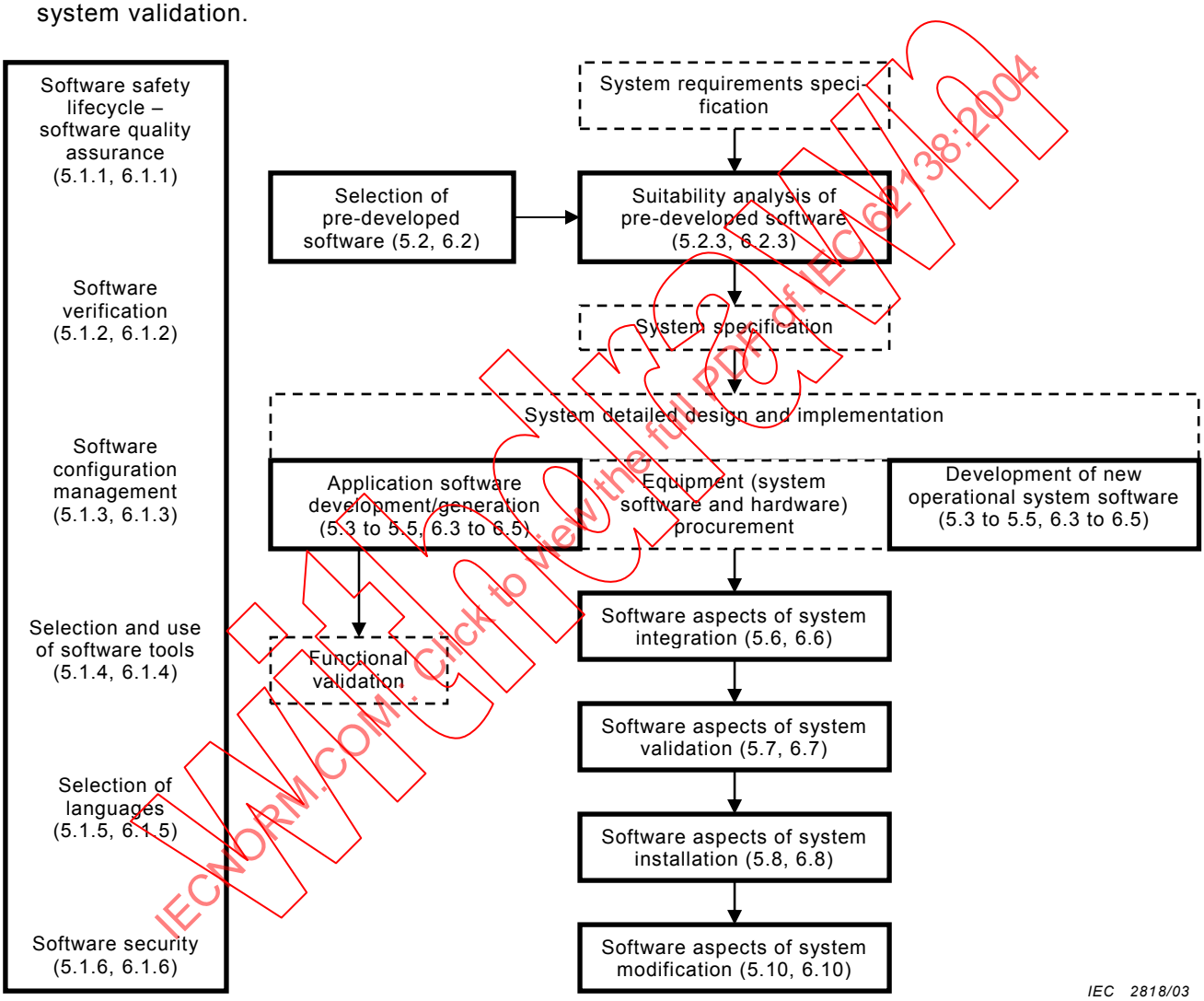

Figure 3 – Software related activities in the System Safety Lifecycle

(boxes in thin dotted lines represent system activities not addressed in this standard)

Les Figures 2 et 3 illustrent les relations entre les activités du Cycle de Vie et de Sûreté du Logiciel et celles du Cycle de Vie et de Sûreté du Système.

Il est à noter que bien que la CEI 61513 identifie deux voies distinctes pour la réalisation de logiciels nouveaux (logiciel d'application et logiciel système opérationnel, voir Figures 2 et 3), cette Norme organise les exigences concernant la réalisation de logiciels nouveaux en quatre paragraphes:

- 5.5.1 et 6.5.1 énoncent des exigences applicables quelle que soit la technique de réalisation utilisée;
- 5.5.2 et 6.5.2 énoncent des exigences propres à la configuration des logiciels pré-développés et des équipements contenant du logiciel, en particulier à la détermination des paramètres et des autres données de configuration;
- 5.5.3 et 6.5.3 énoncent des exigences propres à la réalisation et à la vérification de logiciels en langages orientés application;
- 5.5.4 et 6.5.4 énoncent des exigences propres à la réalisation et à la vérification de logiciels en langages généralistes.

Comme les cases «Développement/Génération du logiciel d'application nouveau» et «Développement du logiciel système opérationnel nouveau» représentent une part importante et essentielle du Cycle de Vie et de Sûreté du Logiciel, un «zoom» est donné en Figure 4, illustrant avec plus de détails les activités entre la spécification des exigences sur le logiciel et la validation du logiciel, avec une représentation claire des trois différentes voies de réalisation (configuration de logiciels et d'équipements pré-développés, utilisation de langages orientés application et utilisation de langages généralistes).

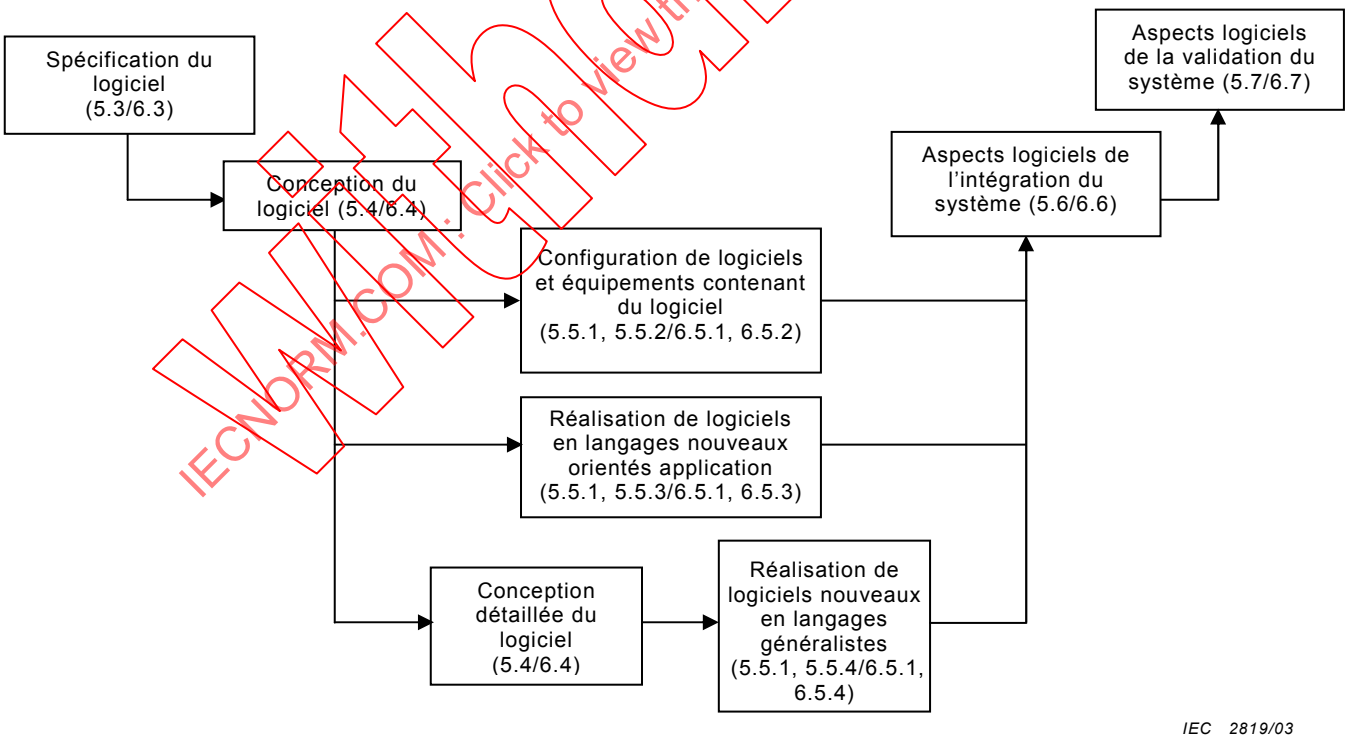

Figure 4 – Activités de développement du Cycle de Vie et de Sûreté du Logiciel selon la CEI 62138

Figures 2 and 3 illustrate the relationship between the activities of the Software Safety Lifecycle and the activities of the System Safety Lifecycle.

It should be noted that although IEC 61513 identifies two different paths for the implementation of new software (application software and operational system software, see Figures 2 and 3), this standard organises the requirements regarding the implementation of new software into four Subclauses:

- 5.5.1 and 6.5.1 provide requirements that are applicable whatever implementation technique is used;
- 5.5.2 and 6.5.2 provide requirements specific to the configuration of pre-developed software and of devices containing software, and in particular the setting of parameters and other configuration data;
- 5.5.3 and 6.5.3 provide requirements specific to the implementation and verification of software in application-oriented languages;
- 5.5.4 and 6.5.4 provide requirements specific to the implementation and verification of software in general-purpose languages.

As boxes titled “Application software development/generation” and “Development of new system software” represent a large and essential part of the Software Safety Lifecycle, a “zoom” is provided in Figure 4, which illustrates in more detail the activities between software requirements specification and software validation, with a clear representation of the three different implementation paths (configuration of pre-developed software and devices, use of application-oriented languages and use of general-purpose languages).

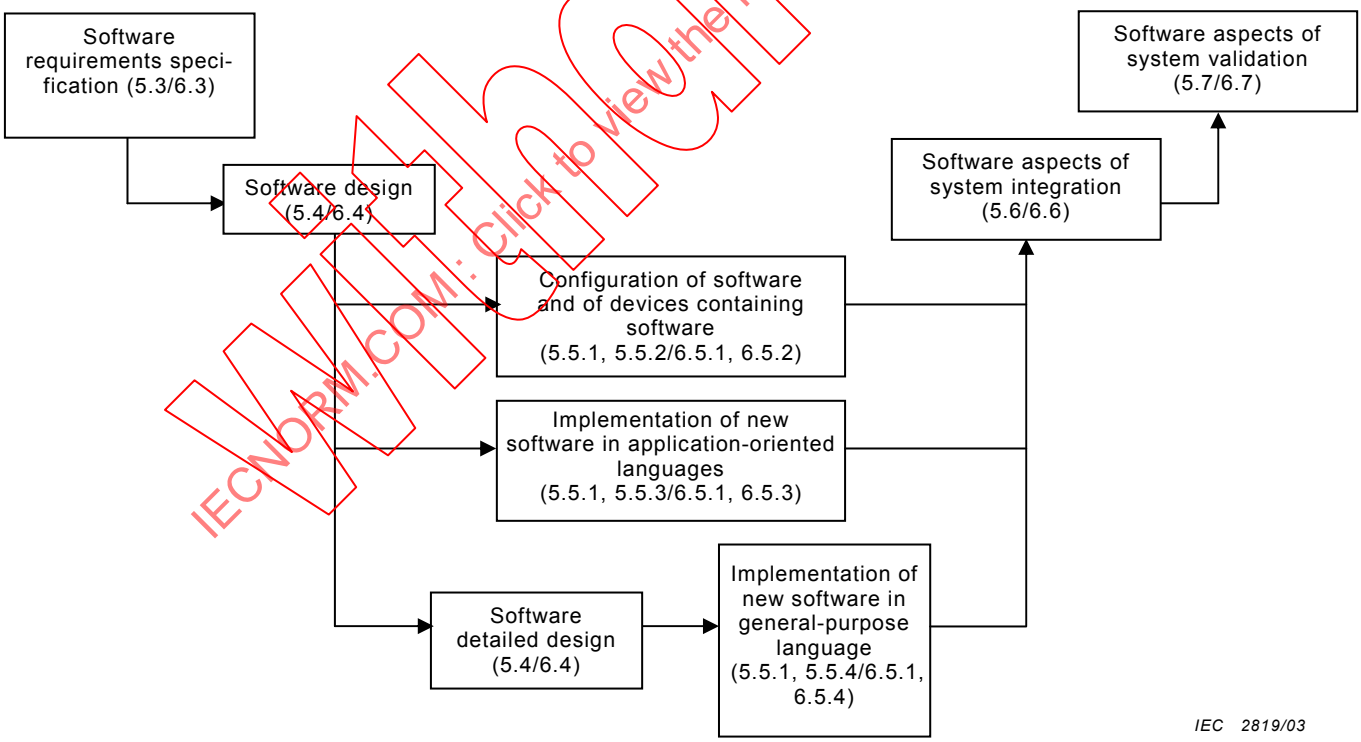

Figure 4 – Development activities of the IEC 62138 Software Safety Lifecycle

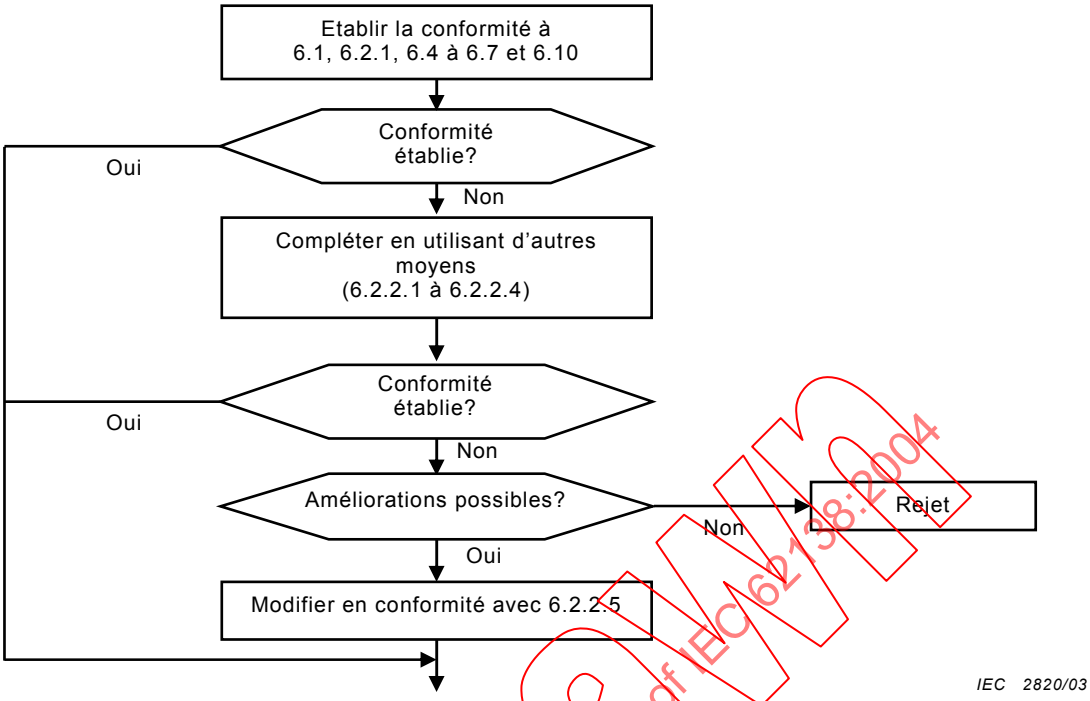

Figure 5 – Processus pour établir que le logiciel pré-développé d'un système d'I&C de classe 2 est correct

Une autre activité du Cycle de Vie et de Sûreté du Logiciel d'une importance particulière est la sélection des logiciels pré-développés, car ces logiciels représentent souvent une portion très significative du logiciel final complet. La Figure 5 illustre en détail le processus de sélection à appliquer pour la classe de sûreté 2.

4.4 Principes de gradation

En conséquence de la gradation de l'importance pour la sûreté des fonctions de catégories A, B et C, une gradation adéquate a été définie pour les exigences applicables aux logiciels des systèmes d'I&C de classes 1, 2 et 3.

Les exigences de cette Norme pour la classe 3 confèrent un niveau de confiance de base pour les logiciels d'un système d'I&C important pour la sûreté. Les principes retenus sont:

- l'appui sur l'Assurance Qualité;
- une attention particulière accordée à l'assurance que les logiciels:
  - contribuent autant que nécessaire aux fonctions de sûreté et n'ont pas d'effet négatif sur elles;
  - sont conformes aux énoncés de spécifications définissant des contraintes importantes pour la sûreté;
- l'assurance que les opérateurs du système d'I&C seront informés aussi tôt que raisonnablement possible des erreurs et défaillances du logiciel susceptibles d'affecter les fonctions identifiées comme importantes pour la sûreté, de façon à permettre toute action appropriée;
- la documentation des exigences, de la conception, de l'intégration, de la validation et de la modification du logiciel.

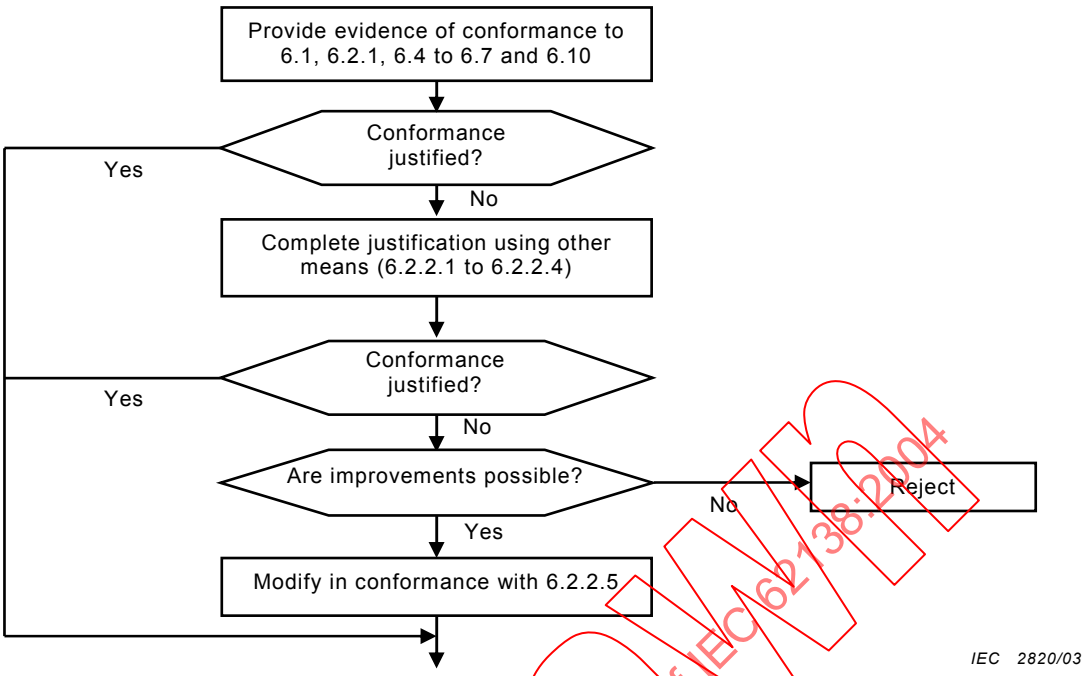

Figure 5 – Process for providing evidence of correctness for pre-developed software of an I&C system of safety class 2

Another activity of particular importance in the Software Safety Lifecycle is the selection of pre-developed software, as this type of software usually represents a very significant portion of the final integrated software. Figure 5 illustrates in more detail the selection process to be used for safety class 2.

4.4 Gradation principles

As a consequence of the gradation of safety relevance for functions of categories A, B and C, a suitable gradation has been adopted for the requirements applicable to the software of I&C systems of safety classes 1, 2 and 3.

The application of the requirements of this standard for safety class 3 confer the basic level of confidence that is suitable for software of an I&C system important to safety. The principles followed are:

- reliance on Quality Assurance;
- special attention given to the assurance that the software:
  - contributes as necessary to, and does not adversely affect, the functions important to safety;
  - satisfies the Software Requirements Specification statements which define constraints important to safety;
- assurance that the operators of the I&C system are informed as early as reasonably possible of software errors and failures that may affect the functions identified as important to safety, so that any appropriate action can be taken;
- documented software requirements specifications, design specifications, integration specifications, validation specifications and modification specifications.

Pour la classe 2, en plus des principes déjà mentionnés pour la classe 3, les principes retenus par la présente Norme sont:

- l'établissement, basé sur des tests et sur la conception, que les propriétés requises pour la sûreté (par exemple les temps de réponse) seront satisfaites dans toutes les conditions spécifiées;
- l'utilisation d'une Documentation pour la Sûreté pour les logiciels pré-développés et les équipements pré-développés contenant du logiciel; l'objectif d'une telle documentation est de fournir toutes les informations nécessaires à une utilisation sûre de ces logiciels et équipements; en particulier, la nécessité d'établir la satisfaction des propriétés requises pour la sûreté détermine le niveau minimum requis de connaissance sur la conception;
- l'utilisation des logiciels pré-développés et des équipements pré-développés contenant du logiciel en fonction de règles s'appuyant sur les Documentations pour la Sûreté correspondantes;
- la configuration et l'utilisation des «boîtes noires» pré-développées en fonction de règles visant également la réduction des effets néfastes des modes de défaillance connus ou supposés;
- l'établissement que les logiciels pré-développés et les équipements pré-développés contenant du logiciel sont corrects et fonctionnellement adéquats; la Figure 5 illustre pour la classe 2 le processus de démonstration qu'un logiciel ou un équipement pré-développé est correct;
- la vérification approfondie et documentée de la conception détaillée et de la réalisation du nouveau logiciel; ceci peut inclure des inspections manuelles, des analyses outillées et des tests;
- des exigences plus sévères pour la vérification, la gestion de configuration, la sélection et l'utilisation des outils logiciels et des langages, la sécurité et la tolérance aux défauts;
- des exigences explicites pour la simplicité, la clarté, la précision, la vérifiabilité, la testabilité et la modifiabilité.

Lorsqu'une exigence est applicable pour les deux classes de sûreté, le niveau de la justification de conformité requis peut dépendre de la classe. En particulier, pour la classe 3, ce niveau peut être moins élevé pour les fonctions non identifiées comme importantes pour la sûreté et ne pouvant pas nuire aux fonctions identifiées comme importantes pour la sûreté.

Les exigences pour les logiciels des systèmes d'I&C de classe 1 sont énoncées par la CEI 60880 et la CEI 60880-2.

Les exigences et recommandations de la présente Norme sont énoncées dans les Articles 5 (pour la classe 3) et 6 (pour la classe 2). Ces Articles peuvent être lus et utilisés indépendamment l'un de l'autre. Ils ont la même structure, de façon que les Articles 5.x.y et 6.x.y énoncent des exigences et des recommandations pour le même sujet. Les exigences de l'Article 6 qui ne sont pas identiques à celles de l'Article 5 (parce qu'elles ont été ajoutées ou modifiées) sont en italique. Toutes les exigences et recommandations sont indentées et numérotées, et, pour une paire d'Articles donnée, celles qui traitent du même problème portent le même numéro. Tous les autres paragraphes sont informatifs.

Il n'est pas dans les intentions de cette Norme de prescrire une forme de documentation particulière. Son but est de définir les informations qui doivent être documentées. La structure et la forme de la documentation peuvent donc varier pourvu que les principes énoncés par la Norme soient respectés.

For safety class 2, in addition to the principles already stated for class 3, the principles followed by this standard are:

- justification, based on tests and design, that the required safety-related performance (for example, response times) will be met in all the specified conditions;
- use of Documentation for Safety for pre-developed software and for pre-developed devices with embedded software; the objective of such documentation is to provide all the information that is necessary for using the software or devices safely; in particular, the need for providing design-based justification regarding the safety-related performance sets the minimal level of information that is necessary;
- use of pre-developed software and devices with embedded software in accordance with rules based on the corresponding Documentations for Safety;
- configuration and use of pre-developed “black boxes” according to rules also aiming at the mitigation of the effects of the known or anticipated failure modes;
- justification of correctness and functional suitability for pre-developed software and pre-developed devices with embedded software; Figure 5 illustrates the process for providing such justification of correctness for safety class 2;
- extensive and documented verification of the detailed design and of the implementation of new software; this may include manual inspections, tool supported analysis and tests;
- more stringent requirements for verification, configuration management, selection and use of software tools and languages, security and fault tolerance;
- explicit requirements for simplicity, clarity, precision, verifiability, testability and modifiability.

When the same requirement is applicable to both safety classes, the extent of the necessary justification of compliance may depend on the safety class. In particular, for class 3, this extent may be less thorough regarding the functions that are not identified as important to safety and that do not jeopardise the functions identified as important to safety.

Requirements for software of I&C systems of safety class 1 are to be found in IEC 60880 and IEC 60880-2.

The requirements and recommendations of this standard are stated in Clauses 5 (for class 3) and 6 (for class 2). These Clauses can be read and used independently from one another. They have the same structure, so that Subclauses 5.x.y and 6.x.y provide requirements and recommendations for the same topic. The requirements of Clause 6 that are not identical to those of Clause 5 (either because they were added or because they were modified) are written in italics. All the requirements and recommendations are indented and numbered, and in a given pair of Subclauses, those addressing the same issue have the same number. All other paragraphs are informative.

It is not the intention of this standard to prescribe a defined set of documentation, but rather to define the information which should be documented. The particular hierarchy and format of documentation adopted may vary, provided that the principles set out in this standard are addressed.

5 Exigences pour le logiciel des systèmes d'I&C réalisant des fonctions de catégorie C[1]

5.1 Exigences générales

5.1.1 Cycle de Vie et de Sûreté du Logiciel – Assurance Qualité du Logiciel

Le paragraphe 6.2.1 de la CEI 61513 énonce des exigences pour l'Assurance Qualité au niveau d'un système d'I&C. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 Le développement du logiciel doit être réalisé selon un Cycle de Vie et de Sûreté du Logiciel. Les dispositions de ce cycle de vie doivent être spécifiées dans un Plan d'Assurance Qualité.

Ce Plan d'Assurance Qualité peut faire partie du Plan d'Assurance Qualité du Système, ou être un Plan d'Assurance Qualité du Logiciel distinct.

2 Si un Plan d'Assurance Qualité du Logiciel est utilisé, il doit être cohérent avec le Plan d'Assurance Qualité du Système. Les exigences applicables de 6.2.1 de la CEI 61513 doivent alors s'appliquer à l'ensemble des deux plans.

3 Le Plan d'Assurance Qualité doit décomposer la phase de développement du Cycle de Vie et de Sûreté du Logiciel en activités spécifiées. Ces activités doivent inclure les activités nécessaires à l'obtention du niveau de qualité spécifié et à la vérification et à la démonstration que cette qualité a été obtenue.

4 La spécification d'une activité doit préciser:

- ses objectifs;
- ses relations et ses interactions avec les autres activités;
- ses entrées et ses résultats;
- l'organisation et les responsabilités correspondantes.

Il convient que le contenu et les propriétés exigés des entrées et des résultats soient également spécifiés.

5 Le Plan d'Assurance Qualité doit exiger que la réalisation de chacune de ces activités soit assignée à des personnes compétentes dotées de ressources adéquates.

6 Le Plan d'Assurance Qualité doit exiger que les modifications de documents déjà approuvés soient identifiées, revues et approuvées par des personnes autorisées.

7 Le Plan d'Assurance Qualité doit exiger que les méthodes, langages, outils, règles et normes utilisés soient identifiés et documentés, et connus et maîtrisés par les personnes concernées.

8 Si plusieurs méthodes, langages, outils, règles et/ou normes sont utilisés, le Plan d'Assurance Qualité doit exiger que ceux qu'il convient d'utiliser pour chaque activité soient clairement identifiés.

9 Le Plan d'Assurance Qualité doit exiger que les termes, expressions, abréviations et conventions utilisés dans un sens spécifique au projet soient explicitement définis.

10 Le Plan d'Assurance Qualité doit exiger que les problèmes rencontrés soient suivis et résolus.

11 Le Plan d'Assurance Qualité doit exiger que des enregistrements résultant de son application soient produits. En particulier, il doit exiger que les résultats des vérifications et revues soient enregistrés avec la nature des contrôles réalisés, les conclusions atteintes et les décisions prises. Les non-conformités au Plan d'Assurance Qualité doivent être documentées et leur justification doit être donnée.

L'ISO 9000-3 donne des conseils supplémentaires pour l'Assurance Qualité du Logiciel.

---

1 La numérotation des points dans les paragraphes suivants correspond à celle de l'Article 6.

5 Requirements for the software of I&C systems performing category C functions[1]

5.1 General requirements

5.1.1 Software Safety Lifecycle – Software Quality Assurance

Subclause 6.2.1 of IEC 61513 provides requirements for Quality Assurance at the level of an I&C system. This Subclause provides additional requirements specific, or of particular importance, to software.

1 The development of software shall be performed according to a Software Safety Lifecycle. The provisions of this Software Safety Lifecycle shall be specified in a Quality Assurance Plan.

This Quality Assurance Plan may be a part of the System Quality Assurance Plan, or may be a separate Software Quality Assurance Plan.

2 If a separate Software Quality Assurance Plan is used, it shall be consistent with the System Quality Assurance Plan. The applicable requirements of 6.2.1 of IEC 61513 shall be addressed by the two plans.

3 The Quality Assurance Plan shall divide the development phase of the Software Safety Lifecycle into specified activities. These activities shall include the activities necessary to achieve the required software quality, and to verify and provide objective evidence that this quality is achieved.

4 The specification of an activity shall state:

- its objectives;
- its relationships and interactions with other activities;
- its inputs and results;
- the organisation and responsibilities relevant to the activity.

The contents and properties required of the inputs and results should also be specified.

5 The Quality Assurance Plan shall require that the implementation of each activity is assigned to competent persons equipped with adequate resources.

6 The Quality Assurance Plan shall require that modifications in already approved documents are identified, reviewed and approved by authorised persons.

7 The Quality Assurance Plan shall require that the methods, languages, tools, rules and standards used are identified and documented, and known to, and mastered by, the persons concerned.

8 The Quality Assurance Plan shall require that if several methods, languages, tools, rules and/or standards are used, it is clear which ones should be used for each activity.

9 The Quality Assurance Plan shall require that project-specific terms, expressions, abbreviations and conventions used are explicitly defined.

10 The Quality Assurance Plan shall require that the issues raised are tracked and resolved.

11 The Quality Assurance Plan shall require that records resulting from its application are produced. In particular, it shall require that the results of verifications and reviews are recorded together with the scope of the verifications or reviews, the conclusions reached and the resolutions agreed. Any deviation from the Quality Assurance Plan shall be documented and justified.

ISO 9000-3 provides additional guidelines for software Quality Assurance.

---

[1] The numbering of the items in the following Subclauses corresponds to that of Clause 6.

5.1.2 Vérification

1 Un Plan de Vérification doit définir la portée des vérifications et des revues devant être réalisées sur le logiciel.

2 Les vérifications et revues doivent être réalisées conformément à des dispositions documentées. En particulier, à des étapes du Cycle de Vie et de Sûreté du Logiciel spécifiées par le Plan de Vérification, on doit vérifier les résultats des activités désignées par le Plan de Vérification afin d'établir que:

- les résultats sont gérés en configuration;
- les activités ont des entrées précisément identifiées, et que les résultats sont cohérents avec ces entrées;
- les activités satisfont aux objectifs spécifiés, que les résultats ont le contenu et les propriétés requis, et qu'ils sont conformes aux décisions prises;
- les résultats sont clairs, précis et à jour;
- les résultats sont conformes aux règles applicables;
- les résultats sont conformes aux exigences applicables de la présente Norme.

«Identification précise» signifie que la version est connue sans ambiguïté. «Clair» signifie que les personnes qui ont à lire un document peuvent le comprendre sans effort excessif même si elles n'ont pas été précédemment impliquées dans le projet, pourvu qu'elles aient les connaissances nécessaires. «Précis» signifie qu'il n'y a pas ambiguïté.

L'étendue des activités de vérification et de revue peut dépendre de la taille et de la nature du logiciel, de la taille et de la nature des résultats vérifiés ou revus, ainsi que des méthodes et outils utilisés. Cette étendue peut aussi être moindre pour les exigences non identifiées comme importantes pour la sûreté (voir l'exigence 6 de 5.3.3) et qui ne peuvent nuire aux fonctions identifiées comme importantes pour la sûreté.

3 La vérification des résultats d'une activité doit être réalisée par des personnes compétentes n'ayant pas participé à cette activité. Il convient d'inclure des représentants de ceux concernés par ces résultats, ainsi que d'autres experts si nécessaire.

Ceci ne signifie pas que l'auteur d'un document ne peut pas être le vérificateur d'un autre.

4 La Spécification du Logiciel, la Documentation de Conception du Logiciel et le Plan de Validation du Logiciel doivent être vérifiés.

5.1.3 Gestion de configuration

Le paragraphe 6.2.1.2 de la CEI 61513 énonce des exigences pour la gestion de configuration au niveau du système d'I&C. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 La gestion de configuration du logiciel doit être réalisée conformément aux dispositions d'un Plan de Gestion de Configuration ou du Plan d'Assurance Qualité. Ces dispositions doivent être cohérentes avec celles du niveau du système.

2 La gestion de configuration doit être appliquée aux éléments permettant d'assurer que le logiciel est correct. Le Plan de Gestion de Configuration doit spécifier quels éléments du logiciel ou quels types d'éléments sont concernés. En particulier, ceci doit inclure:

- les documents clés du Cycle de Vie et de Sûreté du Logiciel (notamment les documents soumis à la vérification);
- les composants logiciels nécessaires à la construction du code exécutable, ainsi que le code exécutable lui-même;

5.1.2 Verification

1 A Verification Plan shall define the scope of software verification and review activities.

2 Verifications and reviews shall be performed according to documented provisions. In particular, at stages of the Software Safety Lifecycle specified by the Verification Plan, the results of activities designated by the Verification Plan shall be verified to show that:

- the results are under configuration management;
- the activities have precisely identified inputs, and their results are consistent with these inputs;
- the activities fulfil their specified objectives, and their results have the required contents and properties, and comply with any resolution agreed;
- the results are clear, precise and up-to-date;
- the results comply with any applicable rule;
- the results comply with the applicable requirements of this standard.

"Precisely identified" means that the version is known without any ambiguity. "Clear" means that the individuals who need to read a document can fully understand it without excessive effort, even if they have not been involved earlier in the project, provided that they have the required knowledge. "Precise" means that there is no ambiguity.

The extent of the verification and review activities may be dependent on the scale and nature of the software, on the scale and nature of the results to be verified or reviewed, and on the methods and tools used. This extent may also be less thorough regarding the specified requirements that are not identified as important to safety (see requirement 6 of 5.3.3) and that cannot jeopardise the functions identified as important to safety.

3 The verification of the results of an activity shall be performed by competent persons who did not participate in the activity. This should include representatives of those concerned with the use of these results, as well as other experts, as necessary.

This does not imply that a person who is an author for one document cannot be the verifier of another.

4 The Software Requirements Specification, the Software Design Specification and the Software Validation Plan shall be verified.

5.1.3 Configuration management

Subclause 6.2.1.2 of IEC 61513 provides requirements for configuration management at the I&C system level. This Subclause provides additional requirements specific, or of particular importance, to software.

1 Configuration management for software shall be performed according to the provisions of a Configuration Management Plan or of the Quality Assurance Plan. These provisions shall be consistent with those for system level configuration management.

2 Configuration management shall be applied to the items related to the correctness of software. The Configuration Management Plan shall specify which software items or types of software items are to be held under configuration management. In particular, these shall include:

- the key documents of the Software Safety Lifecycle (in particular the documents required to be verified);
- the software components necessary to build the executable code, and the executable code itself;

– les outils logiciels permettant d'assurer que le logiciel et/ou la conception du système sont corrects.

3 Le Plan de Gestion de Configuration doit spécifier les moyens techniques permettant l'authentification des éléments du logiciel gérés en configuration, ainsi que de leurs versions.

4 Le Plan de Gestion de Configuration doit assurer une identification non ambiguë de la version du logiciel attachée à une version donnée du système ou d'un équipement, ainsi que des versions de ses éléments constitutifs.

5.1.4 Sélection et utilisation des outils logiciels

Les outils logiciels peuvent jouer un rôle important dans l'évitement des défauts dans le logiciel et dans la conception du système, et dans la mise en évidence des défauts existants. En particulier, des outils peuvent aider ou automatiser la conception de l'architecture des systèmes d'I&C et le développement des logiciels d'application nouveaux.

1 Il convient que des outils logiciels soutiennent les activités de développement qui permettent d'assurer que le logiciel et la conception du système sont corrects.

Il est généralement préférable de ne pas se focaliser uniquement sur la qualité et l'utilisation des outils individuels, mais de prendre également en considération leur compatibilité de façon qu'ils forment un ensemble cohérent. Il est aussi généralement préférable d'utiliser des outils connus bénéficiant d'un retour d'expérience important plutôt que des outils non éprouvés et moins prévisibles. Cependant, chaque situation est à étudier au cas par cas.

2 Il convient que les familles d'équipements utilisées pour le développement d'un système d'I&C soient associées à des outils logiciels capables de réduire le risque d'introduction de défauts dans les logiciels d'application nouveaux.

Ceci comprend en général le support de langages orientés application afin de permettre aux concepteurs de la centrale et de ses systèmes élémentaires de spécifier ou de vérifier les fonctions d'application. L'animation, la génération automatique de code et la génération automatique de cas de test fonctionnels peuvent être également des sujets importants pour de tels outils.

3 Il convient que les familles d'équipement utilisées pour le développement d'un système d'I&C soient associées à des outils logiciels capables de réduire le risque d'introduction de défauts dans la configuration de leurs logiciels pré-développés et dans la conception du système.

Ces outils peuvent par exemple assister le concepteur du système dans:

- l'organisation du système en un ensemble approprié de sous-systèmes interconnectés;
- la répartition des fonctions d'application sur ces sous-systèmes;
- la configuration des sous-systèmes, de leurs communications et de leur logiciel système opérationnel;
- l'assurance que les ressources sont appropriées pour tous les modes de fonctionnement du système;
- la prise en compte des contraintes de conception et de réalisation, en particulier celles visant à ce que le système soit correct et robuste.

4 Le Plan d'Assurance Qualité doit identifier précisément les outils logiciels qui permettent d'assurer que le logiciel et/ou la conception du système sont corrects.

5 Ces outils doivent être accompagnés d'une Documentation d'Utilisation de façon qu'ils soient utilisés comme prévu par leurs concepteurs.

7 Il convient d'établir la qualité et la capacité à produire des résultats corrects des outils qui pourraient introduire des défauts dans le logiciel ou dans la conception du système.

Les générateurs de code et les compilateurs sont des exemples de tels outils.

– the software tools influencing the correctness of software and/or system design.

3 The Configuration Management Plan shall specify technical means for the authentication of the software items under configuration management and of their versions.

4 The Configuration Management Plan shall ensure that the version of the software attached to a given version of the system or equipment, and the versions of the items which together constitute this software version are uniquely identified.

5.1.4 Selection and use of software tools

Software tools can play an important role in preventing the introduction of faults in software or in system design, and in revealing already existing faults. In particular, tools can aid or automate the design of the architecture of I&C systems and the development of new application software.

1 Software tools should support the development activities which contribute to the correctness of software and system design.

It is usually preferable to focus not only on the quality and on the use of individual tools, but also to consider their compatibility with any other tools to be used, so that together, the tools selected form a coherent tool set. Generally it is preferable to use a well-known tool with extensive operational experience rather than an untried tool with no operational experience, but each case needs to be considered on its merits.

2 The equipment families used for the development of an I&C system should be associated with software tools that can reduce the risk of introducing faults in new application software.

These tools usually include support for application-oriented languages, allowing plant and system engineers to specify or verify applications functions. Other significant subjects for such tools may be animation, code generation and aid in the identification of functional test cases.

3 The equipment families used for the development of an I&C system should be associated with software tools that can reduce the risk of introducing faults in the configuration of their pre-developed software and in the design of the system.

Such tools may for example assist system designers in:

- organising the system into a suitable set of interconnected sub-systems;
- distributing the application functions across the sub-systems;
- configuring the sub-systems, their communications and their operational system software;
- ensuring that resources are adequate for all the modes of behaviour of the system;
- taking into account design and implementation constraints, in particular those aiming at the correctness and robustness of the system.

4 The Quality Assurance Plan shall precisely identify the software tools which may influence the correctness of software and/or system design.

5 User Documentation shall be provided for such tools to ensure that they are used as intended.

7 Evidence should be provided regarding the quality of the software tools which might introduce faults in software or in system design, and regarding their ability to produce correct results.

Code generators and compilers are examples of such tools.

L'établissement de la qualité et de la capacité à produire des résultats corrects peut être basé sur le retour d'expérience, la certification des outils, la certification de la qualité des pratiques de développement de leurs fournisseurs, la garantie de l'application de processus de développement et de révision appropriés, et/ou des tests.

5.1.5 Sélection des langages

1 Les langages (orientés application et généralistes) utilisés pour développer le logiciel doivent avoir des syntaxes et des sémantiques précises et documentées.

2 Si des langages orientés application sont disponibles, il convient de leur accorder la préférence.

4 Quand plusieurs langages sont utilisés pour le même code exécutable, les interfaces entre ces langages doivent être documentées.

Les interfaces entre langages incluent par exemple les mécanismes de passation d'arguments et la représentation des structures de données.

6 Il convient que les langages généralistes utilisés supportent le typage statique des variables.

Le typage statique et explicite des variables est recommandé de préférence au typage implicite ou dynamique.

5.1.6 Sécurité

L'objectif de la sécurité est de donner une confiance suffisante dans le fait que les personnes et systèmes non autorisés ne pourront ni modifier le logiciel et ses données ni accéder aux fonctions du système, et dans le fait que cela ne sera pas refusé aux personnes et systèmes autorisés. Les paragraphes 5.4.2 et 6.2.2 de la CEI 61513 énoncent des exigences pour la sécurité au niveau de l'architecture d'I&C et du système d'I&C. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 Une analyse des menaces et des vulnérabilités pour les aspects logiciels de la sécurité du système d'I&C doit être réalisée et documentée. Il convient qu'elle prenne en considération les phases appropriées des Cycles de Vie de Sécurité du Système et du Logiciel. Il convient qu'elle détermine les exigences de protection, d'accessibilité, de confidentialité et d'intégrité des données et des fonctions.

Ceci peut inclure:

- l'identification des données et fonctions critiques pour la sécurité;
- l'identification et l'authentification des personnes;
- le contrôle d'accès aux données et fonctions critiques pour la sécurité;
- la gestion des données et fonctions critiques pour la sécurité;
- la traçabilité des actions liées à la sécurité aux individus.

2 Le développement du logiciel doit être réalisé conformément aux dispositions d'un Plan d'Assurance Sécurité ou d'un Plan d'Assurance Qualité. Ces dispositions doivent prendre en compte les conclusions de l'analyse des menaces et des vulnérabilités. Elles doivent être cohérentes avec les exigences de 5.4.2 et 6.2.2 de la CEI 61513.

3 Le cas échéant, il convient que le logiciel soit configuré et paramétré de façon à limiter les sources de vulnérabilité au strict nécessaire.

4 Il convient que ce Plan inclue des dispositions pour l'évaluation de l'efficacité des solutions mises en oeuvre.

Evidence regarding tool quality and ability to produce correct results may be based on operational experience, tool certification, certification of their suppliers for appropriate development practices, guarantee of appropriate tool development processes, and/or tests.

5.1.5 Selection of languages

1 The languages (application-oriented or general-purpose) used to develop software shall have precise and documented syntax and semantics.

2 Application-oriented languages, if available, should be preferred.

4 When more than one language is used for the same executable code, interfaces between languages shall be documented.

The interface between languages includes argument passing schemes and representation of data structures.

6 The general-purpose languages used should support static typing of variables.

Explicit and static typing of variables is recommended in preference to implicit or dynamic typing.

5.1.6 Security

The objective of security is to provide adequate confidence that unauthorised persons and systems can neither modify the software and its data nor gain access to the system functions, and yet to ensure that this is not denied to authorised persons and systems. Subclauses 5.4.2 and 6.2.2 of IEC 61513 provide requirements for security, at the level of the I&C architecture and of an individual I&C system. This Subclause provides additional requirements specific, or of particular importance, to software.

1 An analysis of the security threats and vulnerability regarding the software aspects of the I&C system shall be performed and documented. It should take into account the relevant phases of the System and Software Safety Lifecycles. It should determine the requirements regarding the protection, the accessibility, the confidentiality and the integrity of data and functions.

These may include:

– identification of security critical data and functions;
– identification and authentication of personnel;
– access control to security critical data and functions;
– management of security critical data and functions;
– traceability of security related actions to personnel.

2 Software development shall be performed according to the provisions of a Security Assurance Plan or of the Quality Assurance Plan. These provisions shall take into account the results of the threat and vulnerability analysis. They shall be consistent with the requirements of 5.4.2 and 6.2.2 of IEC 61513.

3 When relevant, software should be configured and parameterised so as to avoid unnecessary causes of vulnerability.

4 The plan should include provisions for the evaluation of the effectiveness of the solutions implemented.

5.2 Sélection du logiciel pré-développé

Le paragraphe 6.1.2.1 de la CEI 61513 énonce des exigences générales pour la sélection de composants (pas nécessairement logiciels) pré-développés. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

5.2.1 Documentation d'utilisation

5.2.1.1 Objectifs

1 Un logiciel pré-développé doit être accompagné d'une documentation fournissant les informations nécessaires à son utilisation dans le système d'I&C.

Dans la présente Norme, le document ou ensemble de documents correspondant est appelé Documentation d'Utilisation. Quand le logiciel pré-développé fait partie d'un équipement ou d'une famille d'équipements, cette documentation peut être incluse dans la Documentation d'Utilisation de l'équipement ou de la famille d'équipements.

5.2.1.2 Contenu

1 Une Documentation d'Utilisation doit en particulier inclure la description:

- des fonctions offertes;
- des interfaces avec les applications;
- des rôles, types, formats, domaines de valeur et contraintes des entrées, sorties, signaux d'exception, paramètres et données de configuration éventuels;
- des différents modes de fonctionnement et des conditions de transition correspondantes;
- de toute contrainte devant être respectée lors de l'utilisation du logiciel pré-développé.

3 S'il y a lieu, il convient que la Documentation d'Utilisation fournisse également des informations sur les performances des fonctions (les temps de réponse par exemple).

Les fonctions, interfaces et performances peuvent dépendre du mode de fonctionnement, des valeurs des paramètres, des données de configuration et des conditions offertes au logiciel.

5.2.1.3 Propriétés

1 Les énoncés de la Documentation d'Utilisation doivent être précis de façon à éviter des interprétations divergentes.

5.2.2 Conformité à la Documentation d'Utilisation

5.2.2.1 Exigences générales

1 La conformité des logiciels pré-développés en regard des énoncés de leur Documentation d'Utilisation doit être établie.

L'établissement de cette conformité est en général qualitatif, car il n'y a pas de moyen reconnu de tous pour la quantifier. Dans le cas de logiciels pré-développés, la conformité peut être établie sur différents types d'arguments:

- des arguments établissant la conformité à tout ou une partie des exigences de 5.1, 5.2.1, 5.4, 5.5, 5.6, 5.7 et 5.10 s'il y a lieu;
- des tests complémentaires propres au projet;
- des retours d'expérience crédibles et applicables au projet;
- des certifications appropriées par des autorités crédibles.

5.2 Selection of pre-developed software

Subclause 6.1.2.1 of IEC 61513 provides general requirements for the selection of pre-developed components (not necessarily software components). This Subclause provides additional requirements specific, or of particular importance, to software.

5.2.1 User documentation

5.2.1.1 Objectives

1 Pre-developed software shall have documentation giving the information necessary for using the software in the I&C system.

In this standard, the corresponding document or set of documents is called User Documentation. When the pre-developed software is a part of an equipment or equipment family, this documentation may be a part of the User Documentation of the equipment or equipment family.

5.2.1.2 Contents

1 User Documentation shall include a description of:

- the functions provided;
- the interfaces with applications;
- the roles, types, formats, ranges and constraints of inputs, outputs, exception signals, parameters and configuration data, where appropriate;
- the different modes of behaviour and the corresponding conditions of transition;
- any constraint to be respected when using the pre-developed software.

3 When applicable, the User Documentation should also provide information regarding the performances (for example, in terms of response time) of the functions.

Functions, interfaces and performances may depend on the mode of behaviour, on the values of the parameters, on the configuration data and on the conditions provided to the software.

5.2.1.3 Properties

1 User Documentation shall be precise so as to avoid divergent interpretations.

5.2.2 Evidence of correctness

5.2.2.1 General requirements

1 The correctness of pre-developed software with respect to its User Documentation shall be justified.

The justification is usually qualitative because there are no generally recognised means to quantify it. In the case of pre-developed software, it may be based on different types of evidence, for example:

- evidence of conformance to all or part of the requirements of 5.1, 5.2.1, 5.4, 5.5, 5.6, 5.7 and 5.10 when applicable;
- project specific complementary tests;
- applicable and credible operational experience;
- relevant certification by credible authorities.

La confiance peut être plus facile à obtenir lorsqu'un logiciel pré-développé ne peut être utilisé que dans un nombre limité de façons différentes, et/ou lorsque la conception du système d'I&C et de son logiciel garantit des conditions d'utilisation clairement définies.

5.2.2.2 Tests complémentaires

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.2.2.3 Retours d'expérience

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.2.2.4 Certifications

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.2.2.5 Modification

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.2.3 Adéquation fonctionnelle

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.2.4 Sélection et utilisation d'équipements contenant du logiciel

Des équipements en boîte noire contenant du logiciel peuvent être utilisés dans les conditions suivantes:

1 Le logiciel de l'équipement doit lui être intégré de façon à ne pas pouvoir être modifié par l'utilisateur et à ne pas pouvoir être utilisé séparément du reste de l'équipement.

2 Le nombre des fonctions de l'équipement, ses possibilités de configuration et l'étendue de ses interfaces et de ses interactions avec le reste du système d'I&C doivent permettre une bonne couverture fonctionnelle par des tests.

3 Il doit être établi que l'équipement est conforme aux exigences de 5.2.1 et 5.2.2.

4 Il doit être établi que la configuration et l'utilisation de l'équipement dans le système d'I&C sont conformes aux exigences de 5.4, 5.5.1 et 5.5.2.

5.3 Spécification du logiciel

Ce paragraphe complète et précise les exigences de 6.1.2.3 de la CEI 61513.

5.3.1 Objectifs

1 Les exigences sur le logiciel d'un système d'I&C doivent être spécifiées et documentées.

Dans cette Norme, le document ou l'ensemble de documents correspondant est appelé la Spécification du Logiciel. En principe, son objectif est de préciser ce que le logiciel doit accomplir en évitant de spécifier comment le réaliser. Cependant, des contraintes de conception et de réalisation peuvent être spécifiées si elles sont nécessaires compte tenu de la conception du système d'I&C ou de l'architecture d'I&C.

3 La Spécification du Logiciel doit être telle:

- qu'elle contribue à l'établissement que la conception du système d'I&C est correcte;
- que la satisfaction des exigences de la CEI 61513 par le système d'I&C puisse être démontrée.

Confidence may be easier to obtain when the pre-developed software can be used only in a limited number of different ways, and/or when the design of the I&C system and of its software guarantees a well-defined set of conditions of use.

5.2.2.2 Complementary tests

No requirement is necessary for class 3 regarding this subject.

5.2.2.3 Operational experience

No requirement is necessary for class 3 regarding this subject.

5.2.2.4 Certification

No requirement is necessary for class 3 regarding this subject.

5.2.2.5 Modification

No requirement is necessary for class 3 regarding this subject.

5.2.3 Functional suitability

No requirement is necessary for class 3 regarding this subject.

5.2.4 Selection and use of dedicated devices with embedded software

Black-box devices with embedded software may be used in an I&C system under the following conditions.

1 The software of the device shall be integrated in such a way that it cannot be modified by the user and cannot be used separately from the rest of the device.

2 The number of the functions of the device, its potential for configuration, and the extent of its interfaces and interactions with the rest of the I&C system shall be limited so as to allow a thorough functional coverage by tests.

3 Evidence shall be given that the device itself conforms to the requirements of 5.2.1 and 5.2.2.

4 Evidence shall be given that the configuration and the use of the device in the I&C system conforms to the requirements of 5.4, 5.5.1 and 5.5.2.

5.3 Software requirements specification

This Subclause completes and adds precision to the requirements of 6.1.2.3 of IEC 61513.

5.3.1 Objectives

1 The requirements for the software of an I&C system shall be specified and documented.

In this standard, the corresponding document or set of documents is called the Software Requirements Specification. In principle, its objective is to specify what the software is to achieve without specifying how it must do it. However, design and implementation constraints may have to be specified when this is required by considerations of the design of the I&C system or of the I&C architecture.

3 The Software Requirements Specification shall be such that:

- it contributes to the confidence in the correctness of the design of the I&C system;
- compliance of the I&C system to the requirements of IEC 61513 can be demonstrated.

Les exigences de la CEI 61513 concernées par la Spécification du Logiciel sont principalementen 6.1.1.2, 6.1.1.3, 6.1.1.4, 6.1.2.2, 6.1.2.4 et 6.1.3.

4 La Spécification du Logiciel doit être une référence pour la conception et la validation du logiciel, ainsi que pour les modifications éventuelles.

5.3.2 Entrées

3 Les références éventuelles faites par la Spécification du Logiciel à d'autres documents doivent être précises de façon à éviter toute ambiguïté.

5.3.3 Contenu

1 La Spécification du Logiciel doit spécifier:

- les fonctions d'application devant être assurées par le logiciel;
- les différents modes de fonctionnement du logiciel, ainsi que les conditions de transition correspondantes;
- les interfaces et les interactions du logiciel avec son environnement (par exemple avec les opérateurs, avec le reste du système d'I&C, et avec les autres systèmes et équipements avec lesquels il interagit ou partage des ressources), et en particulier les rôles, types, formats, domaines de valeur et contraintes des entrées et des sorties;
- les paramètres du logiciel pouvant être modifiés par les opérateurs en cours d'exploitation, s'il y a lieu, ainsi que leurs rôles, types, formats, domaines de valeur et contraintes, et les contrôles devant être réalisés par le logiciel en cas de modification;
- les performances requises, lorsque cela est pertinent;
- ce que le logiciel ne doit pas faire ou doit éviter, lorsque cela est pertinent;
- les attentes ou les suppositions du logiciel sur son environnement, s'il y a lieu.

2 Il convient que la Spécification du Logiciel spécifie également les conditions que l'environnement offre au logiciel (par exemple les taux de sollicitation), et en particulier les conditions extrêmes.

Les exigences de fonctionnalité, d'interface et de performance peuvent dépendre du mode de fonctionnement, des valeurs des paramètres, des données de configuration, et des conditions offertes au logiciel.

3 La Spécification du Logiciel doit spécifier les modes de fonctionnement du logiciel en cas de détection d'erreur ou de défaillance. Lorsque des tests périodiques sont exigés du système d'I&C, la Spécification du Logiciel doit aussi spécifier le mode de fonctionnement à adopter au cours de ces tests.

4 Il convient que la Spécification du Logiciel précise les objectifs de qualité pour le logiciel. Elle doit préciser les contraintes devant être respectées pour que la conception et la réalisation du logiciel soient correctes et robustes.

Ceci peut inclure des contraintes visant:

- à garantir que le logiciel et la conception du système sont corrects (par exemple des marges dans la gestion des ressources allouées dynamiquement comme la mémoire, la puissance de traitement, la bande passante des canaux de communication et les ressources du système d'exploitation);
- à augmenter la capacité du logiciel et du système d'I&C à tolérer les défauts, à détecter et signaler les erreurs et défaillances, à adopter les modes de fonctionnement spécifiés et à récupérer après une défaillance;
- à garantir que les erreurs des opérateurs et les défaillances des autres systèmes et équipements avec lesquels le logiciel interagit ou partage des ressources n'auront pas de conséquences inacceptables.

The IEC 61513 requirements concerned with Software Requirements Specification are mainly in 6.1.1.2, 6.1.1.3, 6.1.1.4, 6.1.2.2, 6.1.2.4 and 6.1.3.

4 The Software Requirements Specification shall be a reference for software design, software validation, and possible software modifications.

5.3.2 Inputs

3 The references, if any, made by the Software Requirements Specification to other documents shall be precise so as to be unambiguous.

5.3.3 Contents

1 The Software Requirements Specification shall specify:

- the application functions to be provided by the software;
- the different modes of behaviour of the software, and the corresponding conditions of transition;
- the interfaces and interactions of the software with its environment (for example, with operators, with the rest of the I&C system, with the other systems and equipment with which it interacts or shares resources), including the roles, types, formats, ranges and constraints of inputs and outputs;
- the parameters of the software which are to be modified by operators during operation, if any, their roles, types, formats, ranges and constraints, and the checks to be performed by the software when they are modified;
- required performance, when appropriate;
- what the software must not do or must avoid, when appropriate;
- the requirements of, or the assumptions to be made by, the software regarding its environment, when applicable.

2 The Software Requirements Specification should also specify the conditions (for example, the demand load), in particular the worst case conditions, provided to the software by its environment.

Functions, interfaces and performances requirements may depend on the mode of behaviour, on the values of the parameters, on the configuration data and on the conditions provided to the software.

3 The Software Requirements Specification shall specify the software modes of behaviour required when errors or failures are detected. When periodic tests are required of the I&C system, the Software Requirements Specification shall also specify the mode of behaviour required when such tests are performed.

4 The Software Requirements Specification should state the software quality objectives and shall state the constraints to be respected by software design and implementation for the sake of correctness and robustness.

For example, this may include constraints:

- to give confidence in the correctness of software and system design (for example, margins to be taken when using dynamically allocated resources such as memory, processing power, communication bandwidth, operating system resources);
- to enhance the ability of the software and of the I&C system to tolerate faults, to detect and signal errors and failures, to take specified modes of behaviour and to recover from failures;
- to give confidence that mistakes of operators and failures of other systems or equipment with which the software interacts or shares resources will not lead to unacceptable effects.

5 La Spécification du Logiciel doit spécifier la contribution du logiciel à l'assurance que les opérateurs seront informés en temps voulu des erreurs et défaillances concernant les fonctions du système d'I&C identifiées comme importantes pour la sûreté. Les informations délivrées aux opérateurs doivent leur permettre d'entreprendre toute action appropriée.

6 La Spécification du Logiciel doit identifier les fonctions et les exigences relatives à la catégorie de sûreté C.

5.3.4 Propriétés

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.4 Conception du logiciel

5.4.1 Objectifs

1 La conception du logiciel doit être documentée. Il convient que cette documentation donne une vue d'ensemble de la structure et du fonctionnement du logiciel.

Dans cette Norme, le document ou l'ensemble de documents correspondant est appelé la Documentation de Conception du Logiciel. Quand des logiciels pré-développés sont utilisés, la Documentation de Conception du Logiciel peut faire référence aux documentations correspondantes.

3 La Documentation de Conception du Logiciel doit permettre d'établir que les énoncés de la Spécification du Logiciel importants pour la sûreté sont pris en compte et qu'ils seront satisfaits dans toutes les conditions spécifiées.

5 La Documentation de Conception du Logiciel doit garantir, s'il y a lieu, que les effets de bord négatifs des erreurs et défaillances sont réparés avant le retour à un mode de fonctionnement normal.

7 La Documentation de Conception du Logiciel doit être une référence pour la réalisation et l'intégration du logiciel, ainsi que pour les modifications éventuelles.

5.4.2 Entrées

1 Les entrées de la conception du logiciel doivent inclure la Spécification du Logiciel et la Documentation d'Utilisation des logiciels pré-développés.

Il peut aussi y avoir d'autres entrées, par exemple les contraintes spécifiques du projet et/ou les règles et normes applicables.

5.4.3 Contenu

1 La Documentation de Conception du Logiciel doit inclure la spécification:

- de la structure du logiciel;
- du fonctionnement du logiciel dans les conditions et modes de fonctionnement requis par la Spécification du Logiciel.

2 Il convient que la structure du logiciel donne des informations sur:

- l'identification précise et la configuration des logiciels pré-développés;
- la répartition des ressources, des composants logiciels et des tâches logicielles dans les différents sous-systèmes;
- l'allocation des (sous-)fonctions du logiciel et des performances aux tâches logicielles identifiées;
- les principales interfaces, en particulier celles entre tâches logicielles.

5 The Software Requirements Specification shall specify the contribution of the software to the assurance that the operators will be informed in due time of errors or failures concerning the functions of the I&C system identified as important to safety. The information provided to the operators shall allow them to take any appropriate action.

6 The Software Requirements Specification shall identify the functions and the requirements related to safety category C.

5.3.4 Properties

No requirement is necessary for class 3 regarding this subject.

5.4 Software design

5.4.1 Objectives

1 The design of software shall be documented. The documentation should give an overview of the organisation and of the functioning of the software.

In this standard, the corresponding document or set of documents is called the Software Design Specification. When pre-developed software is used, the Software Design Specification may reference the corresponding User Documentation.

3 The Software Design Specification shall provide evidence that the Software Requirements Specification statements important to safety are taken into account and will be satisfied in all specified conditions.

5 The Software Design Specification shall ensure, if applicable, that the adverse side effects of software errors and failures are cleared prior to returning to a normal mode of behaviour.

7 The Software Design Specification shall be a reference for software implementation and integration, and for possible software modifications.

5.4.2 Inputs

1 The inputs to the software design process shall include the Software Requirements Specification and the User Documentation of pre-developed software.

They may also include other documents, such as project specific constraints, and/or applicable rules and standards.

5.4.3 Contents

1 The Software Design Specification shall include the specification of:

- the overall organisation of the software;
- the overall functioning of the software under the conditions and modes of behaviour required by Software Requirements Specification.

2 The overall organisation should provide information regarding:

- the precise identification and the configuration of pre-developed software;
- the distribution of resources, software components and software tasks over sub-systems;
- the allocation of software (sub-)functions and performances to the identified software tasks;
- the main internal interfaces, in particular the interfaces between software tasks.

Il convient que le fonctionnement du logiciel donne des informations sur:

– les interactions, les protocoles de communication et les flux d'informations;
– les ordonnancements et les contraintes temporelles;
– l'utilisation des ressources;
– la synchronisation, en particulier lors de l'utilisation de ressources partagées.

5.4.4 Propriétés

Aucune exigence n'est nécessaire pour la classe 3 sur ce sujet.

5.5 Réalisation du logiciel nouveau

5.5.1 Exigences générales

Les exigences de ce paragraphe sont applicables à tout logiciel nouveau, c'est-à-dire à la configuration des logiciels pré-développés et aux programmes en langages orientés application ou en langages généralistes.

1 Il doit être vérifié que l'utilisation des logiciels pré-développés est conforme aux Documentations d'Utilisation correspondantes et aux contraintes établies par la Documentation de Conception du Logiciel.

2 Les procédures de traduction des programmes nouveaux en code exécutable doivent être documentées et vérifiées.

5.5.2 Configuration des logiciels et des équipements contenant du logiciel

L'exigence de ce paragraphe est spécifique à la configuration des logiciels personnalisables. De tels logiciels peuvent être pré-développés ou nouveaux. L'exigence est également applicable à la configuration des équipements personnalisables contenant du logiciel. Cependant, lorsque les données de configuration représentent des traitements devant être réalisés par le logiciel ou le système (c'est-à-dire lorsqu'elles représentent en réalité des programmes), c'est 5.5.3 qui doit être appliqué.

1 La configuration des logiciels personnalisables et des équipements personnalisables contenant du logiciel doit être documentée.

5.5.3 Réalisation en langages orientés application

Les exigences de ce paragraphe sont spécifiques à la programmation en langages orientés application. En général, des langages orientés application (tels que les diagrammes logiques ou les diagrammes à blocs fonctionnels) peuvent être utilisés pour exprimer tout ou partie de la spécifications ou de la conception du logiciel. L'effort de conception détaillée et de réalisation pour les transformer en programmes pouvant être traduits automatiquement en code exécutable est alors réduit.

1 Les parties de la Spécification du Logiciel et/ou de la Documentation de Conception du Logiciel utilisées pour générer automatiquement du code exécutable doivent être considérées comme des programmes en langages orientés application.

2 L'adéquation fonctionnelle et la cohérence des programmes en langages orientés application réalisant des fonctions importantes pour la sûreté doivent être vérifiées.

The overall functioning should provide information regarding:

- interactions, communication protocols and information flows;
- sequencing and timing constraints;
- use of resources;
- synchronisation, particularly when using shared resources.

5.4.4 Properties

No requirement is necessary for class 3 regarding this subject.

5.5 Implementation of new software

5.5.1 General requirements

The requirements of this Subclause are applicable to all new software, i.e. to the configuration of pre-developed software, and to programs written in application-oriented or general–purpose languages.

1 The use of pre-developed software shall be verified to be consistent with the corresponding User Documentation and with the constraints set by the Software Design Specification.

2 The procedures used to translate new programs into executable code shall be documented and verified.

5.5.2 Configuration of software and of devices containing software

The requirement of this Subclause is specific to the configuration of customisable software. Such software may be pre-developed or new. The requirement is also applicable to the configuration of customisable devices with embedded software. However, when the configuration data represents processing to be performed by the software or the system (i.e. it is effectively software programs), 5.5.3 applies.

1 The configuration of customisable software and devices with embedded software shall be documented.

5.5.3 Implementation with application-oriented languages

The requirements of this Subclause are specific to programs written in application-oriented languages. Generally, application oriented formats (such as logic diagrams or function block diagrams) may be used to express all or part of the Software Requirements Specification or of the Software Design Specification. Only limited detailed design and implementation effort is then necessary to transform the specification into programs that can be automatically translated into executable code.

1 The parts of the Software Requirements Specification and/or of the Software Design Specification that are used to generate executable code by automated means shall be considered to be programs written in application-oriented languages.

2 Programs written in application-oriented languages which are related to functions important to safety shall be verified to be functionally correct and consistent.

5.5.4 Réalisation en langages généralistes

L'exigence de ce paragraphge est spécifique à la programmation en langages généralistes.

4 Les programmes en langages généralistes doivent être réalisés conformément à des règles documentées visant à la clarté, à la modifiabilité et à la testabilité.

Un ensemble de règles peut être propre à un langage ou à un ensemble de programmes.

5.6 Aspects logiciels de l'intégration du système

L'intégration du logiciel est considérée comme faisant partie de l'intégration du système. Le présent paragraphe complète 6.1.4 et 6.2.3 de la CEI 61513 en énonçant des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 L'intégration du logiciel et/ou des inspections doivent établir que le système et le logiciel intégrés:

- sont conformes aux mesures de conception visant à satisfaire les énoncés de la Spécification du Logiciel identifiés comme importants pour la sûreté;
- satisfont aux contraintes énoncées par la Spécification du Logiciel pour que le logiciel soit correct et robuste.

3 L'intégration du logiciel doit être réalisée conformément aux dispositions du Plan d'Intégration du Système ou d'un Plan d'Intégration du Logiciel.

4 Des enregistrements résultant de l'application du plan utilisé pour l'intégration du logiciel doivent être produits (par exemple des résultats des essais). Si des modifications du logiciel ou du système sont nécessaires, il doit être possible de répéter tout ou partie des tests d'intégration pour mettre en évidence les éventuels changements de comportement.

5.7 Aspects logiciels de la validation du système

L'objectif de la validation du logiciel est d'assurer sa conformité aux exigences de fonctionnalité, de performance et d'interface imposées par les exigences sur le système d'I&C. Elle est donc considérée comme faisant partie de la validation du système. Le présent paragraphe complète 6.1.5 et 6.2.4 de la CEI 61513 en énonçant des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 La validation du logiciel doit établir que dans le système d'I&C cible, le logiciel complet est conforme à chacune des exigences de fonctionnalité, de performance et d'interface identifiées comme importantes pour la sûreté. Cela inclut l'établissement que:

- dans les conditions d'utilisation définies par la Spécification du Logiciel, les fonctions logicielles spécifiées et importantes pour la sûreté sont correctement exécutées lorsque les paramètres et les entrées sont dans les domaines de valeur spécifiés;
- dans les conditions d'utilisation définies par la Spécification du Système, les fonctions du système importantes pour la sûreté auxquelles le logiciel contribue sont correctement exécutées;
- le logiciel fournit les protections requises par la Spécification du Logiciel contre les erreurs des opérateurs et les défaillances des autres systèmes et équipements;
- les données d'ingénierie de la centrale utilisées ou intégrées dans le système d'I&C pour réaliser des fonctions importantes pour la sûreté sont correctes; en particulier, la validation du logiciel doit établir que ces données décrivent correctement les systèmes et équipements avec lesquels le logiciel interagit ou partage des ressources.

NOTE L'emploi d'une plateforme matérielle identique à la plateforme cible finale peut être acceptable pour certains tests de validation, pourvu qu'une justification appropriée soit fournie.

Les conditions d'utilisation des fonctions importantes pour la sûreté peuvent inclure l'exécution en parallèle de fonctions non importantes pour la sûreté.

5.5.4 Implementation with general-purpose languages

The requirement of this Subclause is specific to programs written in general-purpose languages.

4 Programs written in general-purpose languages shall conform to documented rules aiming at clarity, modifiability and testability.

A set of rules may be specific to a language or to a set of programs.

5.6 Software aspects of system integration

The integration of software is considered as part of system integration. This Subclause complements 6.1.4 and 6.2.3 of IEC 61513 by providing additional requirements specific, or of particular importance, to software.

1 Software integration and/or inspections shall show that the integrated system and the software:

- comply with the design provisions that ensure the satisfaction of the Software Requirements Specification statements identified as important to safety;
- satisfy the constraints stated by the Software Requirements Specification with respect to correctness and robustness.

3 Software integration shall be performed according to the provisions of the System Integration Plan or of a Software Integration Plan.

4 Records of the application of the plan used for software integration shall be produced, for example, test results. In the event of software or system modifications being required, it shall be possible to repeat all, or a subset of, the integration tests to evaluate the extent of possible changes in behaviour.

5.7 Software aspects of system validation

The objective of the validation of software is to ensure compliance of the integrated software with the functional, performance and interface specifications imposed by the I&C system requirements. Thus, it is considered as part of system validation. This Subclause complements 6.1.5 and 6.2.4 of IEC 61513 by providing additional requirements specific, or of particular importance, to software.

1 Software validation shall show that, in the target I&C system, the integrated software conforms to the functional, performance and interface requirements that are identified as important to safety. This shall include justification that:

- the specified software functions important to safety are correctly performed when their parameters and inputs are in the ranges specified by the Software Requirements Specification, in the conditions of use defined in the Software Requirements Specification;
- the system functions important to safety to which the software contributes are correctly performed in the conditions of use defined in the System Requirements Specification;
- the software provides defences as required by the Software Requirements Specification against errors of operators and failures of other systems and equipment;
- the plant engineering data used by, or integrated in, the I&C system to implement functions important to safety is correct; in particular, the validation of the software shall show that this data correctly describes and addresses the systems and equipment of the plant with which the software interacts or shares resources.

NOTE For some aspects of validation testing, it may be acceptable to use a hardware platform identical to the actual final target platform if adequate justification is provided.

The conditions of use of functions important to safety may include the concurrent operation of functions not important to safety.

2 Il convient que la validation du logiciel soit réalisée conformément aux dispositions du Plan de Validation du Système. Sinon, elle doit être réalisée conformément aux dispositions d'un Plan de Validation du Logiciel.

3 Le plan utilisé pour la validation du logiciel doit spécifier les actions de validation à réaliser, et doit établir que toutes les exigences de fonctionnalité, de performance et d'interface énoncées par la Spécification du Logiciel et importantes pour la sûreté sont bien prises en compte par ces actions. Il doit aussi spécifier les phases principales de la validation du logiciel (par exemple une phase hors site suivie d'une phase sur site) ainsi que les moyens, les méthodes et les outils correspondants.

4 Des enregistrements résultant de l'application du plan utilisé pour la validation du logiciel doivent être produits. Si des modifications du logiciel ou du système sont nécessaires, il doit être possible de répéter tout ou partie des tests de validation pour mettre en évidence les éventuels changements de comportement. Il convient que les résultats de la validation du logiciel soient auditables par des personnes connaissant les sujets traités mais n'ayant pas participé directement au processus de validation.

5 Ces enregistrements doivent décrire la configuration du logiciel validé ainsi que la configuration de l'environnement de validation (par exemple l'environnement matériel et les outils, s'il y a lieu).

6 L'équipe rédigeant le plan utilisé pour la validation du logiciel doit inclure au moins une personne n'ayant pas participé à la conception et à la réalisation.

5.8 Installation du logiciel sur site

Le paragraphe 6.1.6 de la CEI 61513 énonce des exigences relatives à l'installation du système d'I&C sur site. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 La procédure d'installation du logiciel sur site doit être documentée. Elle doit garantir que c'est bien la version correcte et complète du logiciel qui est installée.

2 La procédure d'installation du logiciel sur site doit inclure et spécifier les contrôles à réaliser sur site, ainsi que les tests à réaliser sur le système d'I&C avant son exploitation opérationnelle. En particulier, la satisfaction des conditions requises pour un fonctionnement correct du logiciel doit être vérifiée.

Ces conditions peuvent concerner par exemple le matériel sur lequel s'exécute le logiciel, ou les autres systèmes et équipements avec lesquels le logiciel interagit ou partage des ressources.

5.9 Rapports d'anomalie

1 Il convient qu'un rapport d'anomalie soit établi si un comportement imprévu, apparemment incorrect, inexplicable ou anormal est constaté après la mise en service.

2 Il convient que le rapport d'anomalie précise le comportement observé, la configuration du logiciel et du matériel et les activités en cours au moment du constat. Il convient également de préciser l'auteur, le lieu, la date et l'identification du rapport.

L'identification du rapport peut être donnée après une revue initiale du rapport s'assurant de sa validité.

3 Il convient que les rapports d'anomalie soient passés en revue, et que les problèmes soulevés soient documentés, suivis et résolus.

5.10 Modification du logiciel

Les modifications du logiciel sont décidées en prenant en compte leur impact au niveau du système d'I&C. Elles sont donc soumises aux exigences de 6.1.7 et 6.3.6 de la CEI 61513. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

2 Software validation should be performed according to the provisions of the System Validation Plan. If not, it shall be performed according to the provisions of a Software Validation Plan.

3 The plan used for software validation shall specify the validation actions to be performed, and shall show that all the functionality, performance and interface statements of the Software Requirements Specification identified as important to safety are correctly taken into account by these actions. It shall also specify the main phases of the software validation (for example, an off-site phase followed by an on-site phase) and the corresponding means, methods and tools to be used.

4 Records of the application of the plan used for software validation shall be produced. In the event of software or system modifications being required, it shall be possible to repeat all, or a subset of, the validation tests to evaluate the extent of possible changes in behaviour. The results of software validation should be auditable by persons competent in the subjects addressed but not directly engaged in the validation process.

5 These records shall document the configuration of the software being validated and the configuration of the validation environment (for example, the hardware environment and the tools, if any).

6 The team that writes the plan used for software validation shall include at least one person who did not participate in the design and implementation.

5.8 Installation of software on site

Subclause 6.1.6 of IEC 61513 provides requirements regarding the installation of the I&C system on site. This Subclause provides additional requirements specific, or of particular importance, to the installation of software.

1 The procedure for installing software on site shall be documented. It shall guarantee that the correct and complete version of the software is installed.

2 The procedure for installing software on site shall include and specify on-site checks and tests to be performed before the I&C system is put into full operational use. In particular, the satisfaction of the conditions required for correct operation of the software shall be verified.

For example, these conditions may concern the hardware on which the software operates, or other systems with which the software interacts or shares resources.

5.9 Anomaly reports

1 If unexpected, apparently incorrect, unexplained or abnormal behaviour is experienced after acceptance into service, an anomaly report should be raised.

2 The anomaly report should give details of the behaviour, the software and hardware configurations and the activities in hand at the time. It should also include the originator, location, date, and a report identification.

The report identification may be added following an initial review of the report to ensure that it is valid.

3 The anomaly reports should be reviewed. Issues raised should be documented, tracked and resolved.

5.10 Software modification

The decision to proceed with software modifications depends upon their impact on the I&C system. Therefore, they are subject to the requirements of 6.1.7 and 6.3.6 of IEC 61513. This Subclause provides additional requirements specific, or of particular importance, to software.

1 Les modifications du logiciel doivent être développées de façon à maintenir la conformité aux exigences de 5.1, 5.2, 5.3, 5.4 et 5.5. Elles doivent être installées sur site en conformité avec les exigences de 5.8.

2 Il convient que les modifications du logiciel soient intégrées et validées en conformité avec 5.6 et 5.7. Lorsque l'étendue d'une modification ne nécessite pas l'application complète de ces deux paragraphes, l'intégration du logiciel modifié doit être réalisée selon un Plan de Non-Régression et d'Intégration du Logiciel, et la validation de selon un Plan de Non-Régression et de Validation du Logiciel. La justification de l'adéquation et de la rigueur de ces plans doit être donnée en prenant en compte l'étendue des modifications apportées à la Spécification du Logiciel et à la Documentation de Conception du Logiciel. Des enregistrements résultant de l'application de ces plans doivent être produits.

4 Les modifications du logiciel doivent être documentées de façon détaillée. En particulier, tous les documents logiciels affectés doivent être mis à jour.

7 Les effets d'une modification du logiciel sur le reste du système d'I&C et sur les autres systèmes avec lesquels le logiciel interagit ou partage des ressources doivent être évalués. Toute action nécessaire doit être entreprise afin d'assurer un fonctionnement correct du système d'I&C.

8 Les effets sur le logiciel des modifications dans le reste du système d'I&C ou dans les autres systèmes avec lesquels le logiciel interagit ou partage des ressources doivent être évalués. Toute action nécessaire doit être entreprise afin d'assurer un fonctionnement correct du système d'I&C.

6 Exigences pour le logiciel des systèmes d'I&C réalisant des fonctions de catégorie B

Cet article énonce des exigences supplémentaires pour les logiciels des systèmes d'I&C réalisant des fonctions de catégorie B (donc de classe 2); pour faciliter l'utilisation de la Norme, cet article répète les exigences applicables déjà énoncées pour la classe 3; les exigences supplémentaires ou modifiées sont en italiques.

6.1 Exigences générales

6.1.1 Cycle de Vie et de Sûreté du Logiciel – Assurance Qualité du Logiciel

Le paragraphe 6.2.1 de la CEI 61513 énonce des exigences pour l'Assurance Qualité au niveau d'un système d'I&C. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 Le développement du logiciel doit être réalisé selon un Cycle de Vie et de Sûreté du Logiciel. Les dispositions de ce cycle de vie doivent être spécifiées dans un Plan d'Assurance Qualité.

Ce Plan d'Assurance Qualité peut faire partie du Plan d'Assurance Qualité du Système, ou être un Plan d'Assurance Qualité du Logiciel distinct.

2 Si un Plan d'Assurance Qualité du Logiciel est utilisé, il doit être cohérent avec le Plan d'Assurance Qualité du Système. Les exigences applicables de 6.2.1 de la CEI 61513 doivent alors s'appliquer à l'ensemble des deux plans.

3 Le Plan d'Assurance Qualité doit décomposer la phase de développement du Cycle de Vie et de Sûreté du Logiciel en activités spécifiées. Ces activités doivent inclure les activités nécessaires à l'obtention du niveau de qualité spécifié et à la vérification et à la démonstration que cette qualité a été obtenue.

4 La spécification d'une activité doit préciser:

- ses objectifs;
- ses relations et ses interactions avec les autres activités;
- ses entrées et ses résultats;
- l'organisation et les responsabilités correspondantes.

1 Software modifications shall be developed so as to maintain consistency with the requirements of 5.1, 5.2, 5.3, 5.4 and 5.5. They shall be installed on-site in accordance with the requirements of 5.8.

2 Software modifications should be integrated and validated in a manner consistent with 5.6 and 5.7. When the extent of the modification does not necessitate a full application of these two Subclauses, the integration of the modified software shall be performed according to a Regression Software Integration Plan, and the validation shall be performed according to a Regression Software Validation Plan. The adequacy and thoroughness of these plans shall be justified taking into account the extent of any modifications made in the Software Requirements Specification and in the Software Design Specification. Records of the application of these plans shall be produced.

4 Software modifications shall be comprehensively documented. In particular, all affected software documents shall be updated.

7 The effects of a software modification on the rest of the I&C system and on the other systems with which it interacts or shares resources shall be assessed. Any necessary action shall be taken so as to ensure the correct operation of the I&C system.

8 The effects on software of modifications in the rest of the I&C system or in the other systems with which it interacts or shares resources shall be assessed. Any necessary action shall be taken so as to ensure the correct operation of the I&C system.

6 Requirements for the software of I&C systems performing category B functions

Clause 6 provides additional requirements for software of I&C systems performing category B functions (i.e., systems of safety class 2); in order to facilitate the use of the standard, this clause repeats the relevant requirements for class 3; the additional or modified requirements are in italics.

6.1 General requirements

6.1.1 Software Safety Lifecycle – Software Quality Assurance

Subclause 6.2.1 of IEC 61513 provides requirements for Quality Assurance at the level of an I&C system. This Subclause provides additional requirements specific, or of particular importance, to software.

1 The development of software shall be performed according to a Software Safety Lifecycle. The provisions of this Software Safety Lifecycle shall be specified in a Quality Assurance Plan.

This Quality Assurance Plan may be a part of the System Quality Assurance Plan, or may be a separate Software Quality Assurance Plan.

2 If a separate Software Quality Assurance Plan is used, it shall be consistent with the System Quality Assurance Plan. The applicable requirements of 6.2.1 of IEC 61513 shall be addressed by the two plans.

3 The Quality Assurance Plan shall divide the development phase of the Software Safety Lifecycle into specified activities. These activities shall include the activities necessary to achieve the required software quality, and to verify and provide objective evidence that this quality is achieved.

4 The specification of an activity shall state:

- its objectives;
- its relationships and interactions with other activities;
- its inputs and results;
- the organisation and responsibilities relevant to the activity.

Il convient que le contenu et les propriétés exigés des entrées et des résultats soient également spécifiés.

5 Le Plan d'Assurance Qualité doit exiger que la réalisation de ces activités soit assignée à des personnes compétentes dotées de ressources adéquates.

6 Le Plan d'Assurance Qualité doit exiger que les modifications de documents déjà approuvés soient identifiées, revues et approuvées par des personnes autorisées.

7 Le Plan d'Assurance Qualité doit exiger que les méthodes, langages, outils, règles et normes utilisés soient identifiés et documentés, et connus et maîtrisés par les personnes concernées.

8 Si plusieurs méthodes, langages, outils, règles et/ou normes sont utilisés, le Plan d'Assurance Qualité doit exiger que ceux qu'il convient d'utiliser pour chaque activité soient clairement identifiés.

9 Le Plan d'Assurance Qualité doit exiger que les termes, expressions, abréviations et conventions utilisés dans un sens spécifique au projet soient explicitement définis.

10 Le Plan d'Assurance Qualité doit exiger que les problèmes rencontrés soient suivis et résolus.

11 Le Plan d'Assurance Qualité doit exiger que des enregistrements résultant de son application soient produits. En particulier, il doit exiger que les résultats des vérifications et revues soient enregistrés avec la nature des contrôles réalisés, les conclusions atteintes et les décisions prises. Les non-conformités au Plan d'Assurance Qualité doivent être documentées et leur justification doit être donnée.

L'ISO 9000-3 donne des conseils supplémentaires pour l'Assurance Qualité du Logiciel.

6.1.2 Vérification

1 Un Plan de Vérification doit définir la portée des vérifications et des revues devant être réalisées sur le logiciel.

2 Les vérifications et revues doivent être réalisées conformément à des dispositions documentées. En particulier, à des étapes du Cycle de Vie et de Sûreté du Logiciel spécifiées par le Plan de Vérification, on doit vérifier les résultats des activités désignées par le Plan de Vérification afin d'établir que:

- les résultats sont gérés en configuration;
- les activités ont des entrées précisément identifiées et que les résultats sont cohérents avec ces entrées;
- les activités satisfont les objectifs spécifiés, que les résultats ont le contenu et les propriétés requis, et qu'ils sont conformes aux décisions prises;
- les résultats sont clairs, précis et à jour;
- les résultats sont conformes aux règles applicables;
- les résultats sont conformes aux exigences applicables de la présente Norme.

«Identification précise» signifie que la version est connue sans ambiguïté. «Clair» signifie que les personnes qui ont à lire un document peuvent le comprendre sans effort excessif, même si elles n'ont pas été précédemment impliquées dans le projet, pourvu qu'elles aient les connaissances nécessaires. «Précis» signifie qu'il n'y a pas ambiguïté.

L'étendue des activités de vérification et de revue peut dépendre de la taille et de la nature du logiciel, de la taille et de la nature des résultats vérifiés ou revus, ainsi que des méthodes et outils utilisés.

3 La vérification des résultats d'une activité doit être réalisée par des personnes compétentes n'ayant pas participé à cette activité. Il convient d'inclure des représentants de ceux concernés par ces résultats, ainsi que d'autres experts si nécessaire.

Ceci ne signifie pas que l'auteur d'un document ne peut pas être le vérificateur d'un autre.

The contents and properties required of the inputs and results should also be specified.

5 The Quality Assurance Plan shall require that the implementation of these activities is assigned to competent persons equipped with adequate resources.

6 The Quality Assurance Plan shall require that modifications in already approved documents are identified, reviewed and approved by authorised persons.

7 The Quality Assurance Plan shall require that the methods, languages, tools, rules and standards used are identified and documented, and known to, and mastered by, the persons concerned.

8 The Quality Assurance Plan shall require that if several methods, languages, tools, rules and/or standards are used, it is clear which ones should be used for each activity.

9 The Quality Assurance Plan shall require that project specific terms, expressions, abbreviations and conventions are explicitly defined.

10 The Quality Assurance Plan shall require that the issues raised are tracked and resolved.

11 The Quality Assurance Plan shall require that records resulting from its application are produced. In particular, it shall require that the results of verifications and reviews are recorded together with the scope of the verifications or reviews, the conclusions reached and the resolutions agreed. Any deviation from the Quality Assurance Plan shall be documented and justified.

ISO 9000-3 provides additional guidelines for software Quality Assurance.

6.1.2 Verification

1 A Verification Plan shall define the scope of software verification and review activities.

2 Verifications and reviews shall be performed according to documented provisions. In particular, at stages of the Software Safety Lifecycle specified by the Verification Plan, the results of activities designated by the Verification Plan shall be verified to show that:

- the results are under configuration management;
- the activities have precisely identified inputs, and their results are consistent with these inputs;
- the activities fulfil their specified objectives, and their results have the required contents and properties, and comply with any resolution agreed;
- the results are clear, precise and up-to-date;
- the results comply with any applicable rule;
- the results comply with the applicable requirements of this standard.

“Precisely identified” means that the version is known without any ambiguity. “Clear” means that the individuals who need to read a document can fully understand it without excessive effort, even if they have not been involved earlier in the project, provided that they have the required knowledge. “Precise” means that there is no ambiguity.

The extent of the verification and review activities may be dependent on the scale and nature of the software, on the scale and nature of the results to be verified or reviewed, and on the methods and tools used.

3 The verification of the results of an activity shall be performed by competent persons who did not participate in the activity. This should include representatives of those concerned with the use of these results, as well as other experts, as necessary.

This does not imply that a person who is an author for one document cannot be the verifier of another.

4 La Spécification du Logiciel, la Documentation de Conception du Logiciel et le Plan de Validation du Logiciel doivent être vérifiés.

5 L'application des règles de conception et de réalisation doit être vérifiée.

6.1.3 Gestion de configuration

Le paragraphe 6.2.1.2 de la CEI 61513 énonce des exigences pour la gestion de configuration au niveau du système d'I&C. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 La gestion de configuration du logiciel doit être réalisée conformément aux dispositions d'un Plan de Gestion de Configuration ou du Plan d'Assurance Qualité. Ces dispositions doivent être cohérentes avec celles du niveau du système.

2 La gestion de configuration doit être appliquée aux éléments permettant d'assurer que le logiciel est correct. Le Plan de Gestion de Configuration doit spécifier quels éléments du logiciel ou quels types d'éléments sont concernés. En particulier, ceci doit inclure:

- les documents clés du Cycle de Vie et de Sûreté du Logiciel (notamment les documents soumis à la vérification);
- les composants logiciels nécessaires à la construction du code exécutable, ainsi que le code exécutable lui-même;
- les outils logiciels permettant d'assurer que le logiciel et/ou la conception du système sont corrects.

3 Le Plan de Gestion de configuration doit spécifier les moyens techniques permettant l'authentification des éléments du logiciel gérés en configuration, ainsi que de leurs versions.

4 Le Plan de Gestion de configuration doit assurer une identification non ambiguë de la version du logiciel attachée à une version donnée du système ou d'un équipement, ainsi que des versions de ses éléments constitutifs.

6.1.4 Sélection et utilisation des outils logiciels

Les outils logiciels peuvent jouer un rôle important dans l'évitement des défauts dans le logiciel et dans la conception du système, et dans la mise en évidence des défauts existants. En particulier, des outils peuvent aider ou automatiser la conception de l'architecture des systèmes d'I&C et le développement des logiciels d'application.

1 Il convient que des outils logiciels soutiennent les activités de développement qui permettent d'assurer que le logiciel et la conception du système sont corrects.

Il est généralement préférable de ne pas se focaliser uniquement sur la qualité et l'utilisation des outils individuels, mais de prendre également en considération leur compatibilité de façon qu'ils forment un ensemble cohérent. Il est aussi généralement préférable d'utiliser des outils connus bénéficiant d'un retour d'expérience important plutôt que des outils non éprouvés et moins prévisibles. Cependant, chaque situation est à étudier au cas par cas.

2 Les familles d'équipements utilisées pour le développement d'un système d'I&C doivent être associées à des outils logiciels capables de réduire le risque d'introduction de défauts dans les logiciels d'application nouveaux.

Ceci comprend en général le support de langages orientés application afin de permettre aux concepteurs de la centrale et de ses systèmes élémentaires de spécifier ou de vérifier les fonctions d'application. L'animation, la génération automatique de code et la génération automatique de cas de test fonctionnels peuvent être également des sujets importants pour de tels outils.

3 Il convient que les familles d'équipement utilisées pour le développement d'un système d'I&C soient associées à des outils logiciels capables de réduire le risque d'introduction de défauts dans la configuration de leurs logiciels pré-développés et dans la conception du système.

4 The Software Requirements Specification, the Software Design Specification and the Software Validation Plan shall be verified.

5 The application of design and implementation rules shall be verified.

6.1.3 Configuration Management

Subclause 6.2.1.2 of IEC 61513 provides requirements for configuration management at the I&C system level. This Subclause provides additional requirements specific, or of particular importance, to software.

1 Configuration management for software shall be performed according to the provisions of a Configuration Management Plan or of the Quality Assurance Plan. These provisions shall be consistent with those for system level configuration management.

2 Configuration management shall be applied to the items related to the correctness of software. The Configuration Management Plan shall specify which software items or types of software items are to be held under configuration management. In particular, these shall include:

- the key documents of the Software Safety Lifecycle (in particular the documents required to be verified);
- the software components necessary to build the executable code, and the executable code itself;
- the software tools influencing the correctness of software and/or system design.

3 The Configuration Management Plan shall specify technical means for the authentication of the software items under configuration management and of their versions.

4 The Configuration Management Plan shall ensure that the version of the software attached to a given version of the system or equipment, and the versions of the items which together constitute this software version are uniquely identified.

6.1.4 Selection and use of software tools

Software tools can play an important role in preventing the introduction of faults in software or in system design, and in revealing existing faults. In particular, tools can aid or automate the design of the architecture of I&C systems and the development of new application software.

1 Software tools should support the development activities which contribute to the correctness of software and system design.

It is usually preferable to focus not only on the quality and on the use of individual tools, but also to consider their compatibility with any other tools to be used, so that together, the tools selected form a coherent tool set. Generally it is preferable to use a well-known tool with extensive operational experience rather than an untried tool with no operational experience, but each case needs to be considered on its merits.

2 The equipment families used for the development of an I&C system shall be associated with software tools that can reduce the risk of introducing faults in new application software.

These tools usually include support for application-oriented languages, allowing plant and system engineers to specify or verify applications functions. Other significant subjects for such tools may be animation, code generation and aid in the identification of functional test cases.

3 The equipment families used for the development of an I&C system should be associated with software tools that can reduce the risk of introducing faults in the configuration of their pre-developed software and in the design of the system.

Ces outils peuvent par exemple assister le concepteur du système dans:

– l'organisation du système en un ensemble approprié de sous-systèmes interconnectés;
– la répartition des fonctions d'application sur ces sous-systèmes;
– la configuration des sous-systèmes, de leurs communications et de leur logiciel système opérationnel;
– l'assurance que les ressources sont appropriées pour tous les modes de fonctionnement du système;
– la prise en compte des contraintes de conception et de réalisation, en particulier celles visant à ce que le système soit correct et robuste.

4 Le Plan d'Assurance Qualité doit identifier précisément les outils logiciels qui permettent d'assurer que le logiciel et/ou la conception du système sont corrects.

5 Ces outils doivent être accompagnés d'une Documentation d'Utilisation de façon qu'ils soient utilisés comme prévu par leurs concepteurs.

6 Le Plan d'Assurance Qualité doit distinguer les outils qui pourraient introduire des défauts dans le logiciel ou dans la conception du système, de ceux qui pourraient seulement conduire à ignorer des défauts déjà présents.

Les générateurs de code et les compilateurs sont des exemples d'outils de la première catégorie, alors que les analyseurs de code et les générateurs de cas de test sont des exemples de la seconde catégorie.

7 Les outils logiciels qui pourraient introduire des défauts dans le logiciel ou dans la conception du système doivent être sélectionnés et utilisés conformément à des procédures et règles visant à réduire ce risque. Leur qualité et leur capacité à produire des résultats corrects doivent être établies. Leur utilisation doit être tracée de façon qu'il soit possible d'identifier les outils ayant permis de produire une information donnée.

L'établissement de la qualité et de la capacité à produire des résultats corrects peut être basé sur le retour d'expérience, la certification ou la qualification des outils, la certification de la qualité des pratiques de développement de leurs fournisseurs, la garantie de l'application de processus de développement et de révision appropriés, et/ou des tests. *La rigueur de la démonstration peut dépendre des conditions d'utilisation de l'outil, de l'intensité de la vérification des résultats, de la probabilité que des résultats erronés ne soient pas détectés, et de la gravité des conséquences des résultats erronés non détectés. Inversement, une démonstration rigoureuse (par exemple une qualification selon la CEI 60880-2) peut se substituer à certaines vérifications des résultats.*

8 Il convient que les outils logiciels qui pourraient conduire à ignorer des défauts déjà présents dans le logiciel ou dans la conception du système soient sélectionnés et utilisés de façon à réduire ce risque. Il convient que leur utilisation soit tracée.

9 Quand un outil ou une version d'outil susceptible d'introduire des défauts dans le logiciel ou dans la conception du système est remplacé(e) par un(e) autre, des précautions raisonnables doivent être prises afin d'établir que cela n'aura d'effet négatif ni sur le logiciel, ni sur la conception du système.

Par exemple, en plus de la qualité et de la capacité à produire des résultats corrects, la compatibilité avec l'outil précédent peut devoir être analysée.

6.1.5 Sélection des langages

1 Les langages (orientés application et généralistes) utilisés pour développer le logiciel doivent avoir des syntaxes et des sémantiques précises et documentées.

2 Si des langages orientés application sont disponibles, il convient de leur accorder la préférence.

Such tools may for example assist system designers in:

- organising the system into a suitable set of interconnected sub-systems;
- distributing the application functions across the sub-systems;
- configuring the sub-systems, their communications and their operational system software;
- ensuring that resources are adequate for all the modes of behaviour of the system;
- taking into account design and implementation constraints, in particular those aiming at the correctness and robustness of the system.

4 The Quality Assurance Plan shall precisely identify the software tools which may influence the correctness of software and/or system design.

5 User Documentation shall be provided for such tools to ensure that they are used as intended.

6 The Quality Assurance Plan shall distinguish the tools which might introduce faults in software or in system design, from those which only might lead to overlooking already existing faults.

Code generators and compilers are examples of tools of the first category, whereas static code analysers and test case generators are examples of tools of the second category.

7 The software tools which might introduce faults in software or in system design shall be selected and used according to documented procedures and rules aiming at reducing this risk. Evidence shall be provided regarding their quality and their ability to produce correct results. Their use shall be traced so that the tools, if any, which were used to generate a given item or information may be identified.

Evidence regarding tool quality and ability to produce correct results may be based on operational experience, tool qualification or certification, certification of their suppliers for appropriate development practices, guarantee of appropriate tool development processes, and/or tests. *The stringency of the evidence may depend on the conditions of use of the tool, on the extent of the verification of its outputs, on the likelihood of tool errors to be detected, and on the seriousness of the consequences of undetected erroneous results. Conversely, stringent evidence (for example a tool qualification according to IEC 60880-2) may be used as a substitute for some of the verifications of outputs.*

8 The software tools which might lead to faults already existing in software or in system design being overlooked should be selected and used in a way which reduces this risk. Their use should be traced.

9 When a tool or tool version which has the potential to introduce faults in software or in system design is substituted with another, reasonable precautions shall be taken to ensure that this does not have adverse effects on the correctness of the software and the system design.

For example, in addition to the quality and ability of the new tool to produce correct results, its compatibility with the previous tool may need to be assessed.

6.1.5 Selection of languages

1 The languages (application-oriented or general-purpose) used to develop software shall have precise and documented syntax and semantics.

2 Application-oriented languages, if available, should be preferred.

3 *Les langages généralistes de bas niveau orientés machine (les langages d'assemblage par exemple) peuvent être utilisés pour des programmes particuliers, mais il convient de le justifier.*

4 Quand plusieurs langages sont utilisés pour le même code exécutable, les interfaces entre ces langages doivent être documentées.

Les interfaces entre langages incluent par exemple les mécanismes de passation d'arguments et la représentation des structures de données.

5 *Il convient que les langages généralistes utilisés aient des caractéristiques facilitant l'analyse statique des programmes par des outils.*

6 En particulier, il convient que les langages généralistes utilisés supportent le typage statique des variables. *Il convient d'expliciter statiquement le type des variables plutôt que d'utiliser un typage implicite ou dynamique.*

7 *Les langages utilisés et leurs bibliothèques de fonctions doivent permettre un comportement prédictible du logiciel.*

Par exemple, la perturbation du comportement normal du logiciel à des moments aléatoires pour récupérer la mémoire libérée n'est en général pas acceptable.

6.1.6 Sécurité

L'objectif de la sécurité est de donner une confiance suffisante dans le fait que les personnes et systèmes non autorisés ne pourront ni modifier le logiciel et ses données ni accéder aux fonctions du système, et dans le fait que cela ne sera pas refusé aux personnes et systèmes autorisés. Les paragraphes 5.4.2 et 6.2.2 de la CEI 61513 énoncent des exigences pour la sécurité au niveau de l'architecture d'I&C et du système d'I&C. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

1 Une analyse des menaces et des vulnérabilités pour les aspects logiciels de la sécurité du système d'I&C doit être réalisée et documentée. *Elle doit prendre en considération les phases appropriées des Cycles de Vie de Sécurité du Système et du Logiciel. Elle doit déterminer les exigences de protection, d'accessibilité, de confidentialité et d'intégrité des données et des fonctions.*

Ceci peut inclure:

- l'identification des données et fonctions critiques pour la sécurité;
- l'identification et l'authentification des personnes;
- le contrôle d'accès aux données et fonctions critiques pour la sécurité;
- la gestion des données et fonctions critiques pour la sécurité;
- la traçabilité des actions liées à la sécurité des individus.

2 Le développement du logiciel doit être réalisé conformément aux dispositions d'un Plan d'Assurance Sécurité ou d'un Plan d'Assurance Qualité. Ces dispositions doivent prendre en compte les conclusions de l'analyse des menaces et des vulnérabilités. Elles doivent être cohérentes avec les exigences de 5.4.2 et 6.2.2 de la CEI 61513.

3 Le cas échéant, il convient que le logiciel soit configuré et paramétré de façon à limiter les sources de vulnérabilité au strict nécessaire.

4 *Le plan utilisé doit inclure des dispositions pour l'évaluation de l'efficacité des solutions mises en œuvre.*

6.2 Sélection des logiciels prédéveloppés

Le paragraphe 6.1.2.1 de la CEI 61513 énonce des exigences générales pour la sélection de composants (pas nécessairement logiciels) pré-développés. Le présent paragraphe énonce des exigences complémentaires spécifiques ou d'une importance particulière pour le logiciel.

3 Low level, machine-oriented general-purpose languages (for example, assembly languages) may be used for specific software programs, but this should be justified.

4 When more than one language is used for the same executable code, interfaces between languages shall be documented.

The interface between languages includes argument passing schemes and representation of data structures.

5 The general-purpose languages used should have features facilitating tool supported static analyses of programs.

6 In particular, the general-purpose languages used should support static typing of variables. *Explicit and static typing of variables should be used in preference to implicit or dynamic typing.*

7 The languages used and their corresponding run-time libraries shall enable predictable run-time behaviour of the software.

For example, disruption of the normal behaviour of the software for the collection of freed memory at random moments is usually not acceptable.

6.1.6 Security

The objective of security is to provide adequate confidence that unauthorised persons and systems can neither modify the software and its data nor gain access to the system functions, and yet to ensure that this is not denied to authorised persons and systems. Subclauses 5.4.2 and 6.2.2 of IEC 61513 provide requirements for security, at the level of the I&C architecture and of an individual I&C system. This Subclause provides additional requirements specific, or of particular importance, to software.

1 An analysis of the security threats and vulnerability regarding the software aspects of the I&C system shall be performed and documented. *It shall take into account the relevant phases of the System and Software Safety Lifecycles. It shall determine the requirements regarding the protection, the accessibility, the confidentiality and the integrity of data and functions.*

These may include:

- identification of security critical data and functions;
- identification and authentication of personnel;
- access control to security critical data and functions;
- management of security critical data and functions;
- traceability of security related actions to personnel.

2 Software development shall be performed according to the provisions of a Security Assurance Plan or of the Quality Assurance Plan. These provisions shall take into account the results of the threats and vulnerability analysis. They shall be consistent with the requirements of 5.4.2 and 6.2.2 of IEC 61513.

3 When relevant, software should be configured and parameterised so as to avoid unnecessary causes of vulnerability.

4 The Plan shall include provisions for the evaluation of the effectiveness of the solutions implemented.

6.2 Selection of pre-developed software

Subclause 6.1.2.1 of IEC 61513 provides general requirements for the selection of pre-developed components (not necessarily software components). This Subclause provides additional requirements specific, or of particular importance, to software.

6.2.1 Documentation pour la Sûreté

6.2.1.1 Objectifs

1 Un logiciel pré-développé doit être accompagné d'une documentation fournissant les informations nécessaires à une utilisation sûre dans le système d'I&C et à l'établissement de la conformité aux exigences de 6.2.3 et 6.4.

Dans la présente Norme, le document ou ensemble de documents correspondant est appelé Documentation pour la Sûreté. Quand le logiciel pré-développé fait partie d'un équipement ou d'une famille d'équipements, cette documentation peut être incluse dans la Documentation pour la Sûreté de l'équipement ou de la famille d'équipements.

Une Documentation pour la Sûreté peut être générique ou être spécifique à un projet. Elle peut comprendre plus que la Documentation d'Utilisation délivrée par le fournisseur. Par exemple, elle peut inclure des informations obtenues par des essais, mesures, et/ou analyses complémentaires, ou par des retours d'expérience.

6.2.1.2 Contenu

1 Une Documentation pour la Sûreté doit en particulier inclure la description:

- des fonctions offertes;
- des interfaces avec les applications;
- des rôles, types, formats, domaines de valeur et contraintes des entrées, sorties, signaux d'exception, paramètres et données de configuration éventuels;
- des différents modes de fonctionnement et des conditions de transition correspondantes;
- de toute contrainte devant être respectée lors de l'utilisation du logiciel pré-développé.

2 Le cas échéant, il convient que ces contraintes visent à:

- *donner confiance dans le fait que le logiciel complet et la conception du système sont corrects (par exemple des marges devant être prises dans l'utilisation des ressources allouées dynamiquement comme la mémoire, la puissance de calcul, la bande passante des moyens de communication et les ressources du système d'exploitation);*
- *améliorer la capacité du logiciel complet et du système d'I&C à détecter, signaler et tolérer les défaillances, à adopter les modes de fonctionnement spécifiés et à récupérer après une défaillance;*
- *donner confiance dans le fait que les erreurs des opérateurs et les défaillances des autres systèmes et équipements interagissant ou partageant des ressources avec le logiciel complet conduiront à des modes de fonctionnement définis;*
- *garantir que l'environnement du logiciel pré-développé lui offrira toutes les ressources nécessaires dans toutes les conditions d'utilisation au sein du système d'I&C.*

3 S'il y a lieu, il convient que la Documentation pour la Sûreté fournisse également des informations sur les performances des fonctions (les temps de réponse par exemple).

Les fonctions, interfaces et performances peuvent dépendre du mode de fonctionnement, des valeurs des paramètres, des données de configuration et des conditions offertes au logiciel.

4 La Documentation pour la Sûreté doit aussi fournir des informations sur:

- *l'auto surveillance mise en oeuvre, les capacités de tolérance aux défauts et les modes de défaillance;*
- *les exigences du logiciel pré-développé vis-à-vis de son environnement (par exemple vis-à-vis du matériel ou des autres composants logiciels);*
- *les interactions et les interfaces du logiciel pré-développé avec le matériel nécessaires pour déterminer un fonctionnement sûr du système.*

6.2.1 Documentation for Safety

6.2.1.1 Objectives

1 Pre-developed software shall have documentation giving the information necessary for using the software safely in the I&C system and for providing evidence that the requirements of 6.2.3 (Functional suitability) and 6.4 (Software design) are satisfied.

In this standard, the corresponding document or set of documents is called Documentation for Safety. When the pre-developed software is a part of an equipment or equipment family, this documentation may be a part of the Documentation for Safety of the equipment or equipment family.

A Documentation for Safety may be generic or project specific. It may comprise more than the User Documentation provided by the supplier of the pre-developed software. For example, it may include information obtained from additional tests, measurements and/or analyses, and from operational experience.

6.2.1.2 Contents

1 Documentation for Safety shall include a description of:

- the functions provided;
- the interfaces with applications;
- the roles, types, formats, ranges and constraints of inputs, outputs, exception signals, parameters and configuration data, where appropriate;
- the different modes of behaviour and the corresponding conditions of transition;
- any constraint to be respected when using the pre-developed software.

2 When applicable, these constraints should aim at:

- *giving confidence in the correctness of the integrated software and of the system design (for example, margins to be taken when using dynamically allocated resources such as memory, processing power, communication bandwidth, operating system resources);*
- *enhancing the ability of the integrated software and of the I&C system to detect, signal and tolerate failures, to take specified modes of behaviour and to recover from failures;*
- *giving confidence that mistakes of operators and failures of other systems or equipment with which the integrated software interacts or shares resources will lead to defined modes of behaviour;*
- *guaranteeing that the environment of the pre-developed software will provide all the necessary resources in all the conditions of use in the I&C system.*

3 When applicable, the Documentation for Safety should also provide information regarding the performances (for example, in terms of response time) of the functions.

Functions, interfaces and performances may depend on the mode of behaviour, on the values of the parameters, on the configuration data and on the conditions provided to the software.

4 The Documentation for Safety shall also provide information regarding:

- *the self-surveillance performed, the fault tolerance capability and the failure modes;*
- *the requirements of the pre-developed software regarding its environment (for example, regarding hardware or other software components);*
- *the interactions and interfaces of the pre-developed software with the hardware, to the extent necessary to fully define the safe functional performance of the system.*

5 La Documentation pour la Sûreté du logiciel système opérationnel d'une famille d'équipements pré-développée doit fournir des informations permettant, pour des applications conformes, de prévoir les caractéristiques clés d'un fonctionnement du système, telles que les temps de réponse maximaux et les besoins maximaux en ressources.

De telles informations peuvent être fournies sous la forme de données, formules et/ou modèles permettant le calcul de majorants du temps de réponse et des ressources utilisées. Lorsque le logiciel offre une large gamme de fonctions, interfaces et possibilités de configuration, une confiance suffisante dans ces informations peut être difficile à obtenir sans la connaissance du fonctionnement du logiciel.

6.2.1.3 Propriétés

1 Les énoncés de la Documentation pour la Sûreté doivent être précis de façon à éviter des interprétations divergentes.

6.2.2 Conformité à la Documentation pour la Sûreté

6.2.2.1 Exigences générales

1 La conformité des logiciels pré-développés en regard des énoncés de leur Documentation pour la Sûreté doit être établie.

L'établissement de cette conformité est en général qualitatif, car il n'y a pas de moyen reconnu de tous pour la quantifier. *La Figure 5 illustre les approches envisageables:*

- *quand un logiciel pré-développé a été développé selon 6.1, 6.2.1, 6.4, 6.5, 6.6, 6.7 et, s'il y a lieu, 6.10, aucune autre démonstration n'est nécessaire;*
- *quand ceci ne peut être suffisamment établi, les paragraphes 6.2.2.2, 6.2.2.3 et 6.2.2.4 offrent des moyens complémentaires pour terminer la démonstration.*

2 Lorsque des moyens complémentaires sont utilisés, il convient que les critères d'acceptation soient spécifiés et que justification en soit donnée dans les phases amont du Cycle de Vie et de Sûreté du Logiciel. Il convient que ces critères soient justifiés par en regard des exigences de cette Norme pour lesquelles la conformité n'a pas été suffisamment établie.

La confiance peut être plus facile à obtenir lorsqu'un logiciel pré-développé ne peut être utilisé que dans un nombre limité de façons différentes, et/ou lorsque la conception du système d'I&C et de son logiciel garantit des conditions d'utilisation clairement définies.

6.2.2.2 Tests complémentaires

1 Les tests réalisés sur un logiciel pré-développé durant le développement du système d'I&C doivent être documentés afin de compléter la démonstration de conformité doivent être documentés. Ils doivent démontrer que dans les conditions au sein du système d'I&C, ce logiciel est et se comporte comme énoncé dans sa Documentation pour la Sûreté.

Ces conditions peuvent concerner la configuration du logiciel pré-développé (en particulier les paramètres et des données de configuration), l'utilisation des fonctions et interfaces, l'environnement matériel, le processeur et le taux de sollicitation.

2 Il convient que les règles d'élaboration des tests complémentaires soient documentées. Il convient aussi d'en donner la justification.

3 La documentation des tests complémentaires doit préciser:

- *la version du logiciel concernée, et s'il y a lieu, la ou les configurations du logiciel pré-développé;*
- *une description des tests effectués, et s'il y a lieu, de l'environnement utilisé, de façon à pouvoir répéter les tests dans des conditions identiques;*

5 The Documentation for Safety of the operational system software of a pre-developed equipment family shall provide information enabling correct predictions regarding the key safety significant elements of system performance, for example, the maximum response times of, or the maximum usage of resources by, compliant applications.

Such information may be provided in the form of data, formulae and/or models allowing the calculation of worst case response times and resources usage of applications. When the software offers a wide range of functions, interfaces and possibilities for configuration, an appropriate confidence in the correctness of the information may be difficult to obtain without knowledge of the functioning of the software.

6.2.1.3 Properties

1 Documentation for Safety shall be precise so as to avoid divergent interpretations.

6.2.2 Evidence of correctness

6.2.2.1 General requirements

1 The correctness of pre-developed software with respect to its Documentation for Safety shall be justified.

The justification is usually qualitative because there are no generally recognised means to quantify it. *Figure 5 illustrates the approaches that can be taken:*

- *when pre-developed software has been developed in conformance to the applicable requirements of 6.1, 6.2.1, 6.4, 6.5, 6.6, 6.7 and 6.10 when applicable, no additional justification of correctness is necessary;*
- *when conformance to these requirements cannot be adequately justified, then 6.2.2.2, 6.2.2.3 and 6.2.2.4 provide complementary means which may be used to complete the justification.*

2 When using complementary means for providing evidence of correctness, the acceptance criteria should be specified and justified in early stages of the Software Safety Lifecycle. These criteria should be justified considering the requirements of this standard the compliance to which has not been adequately established.

Confidence may be easier to obtain when a pre-developed software can be used only in a limited number of different ways, and/or when the design of the I&C system and of its software guarantees clearly defined conditions of use.

6.2.2.2 *Complementary tests*

1 The tests performed on a pre-developed software during the development of an I&C system shall be documented. They shall provide evidence that, in the conditions of use within the I&C system, this pre-developed software is and behaves as specified by its Documentation for Safety.

The conditions of use may concern aspects such as the configuration of the pre-developed software (particularly the setting of parameters and configuration data), the use of functions and interfaces, the hardware environment, the processor and the demand loads.

2 The rules used to design complementary tests should be documented and justified.

3 The documentation of complementary tests shall record:

- *the version concerned and, when relevant, the configuration of the pre-developed software;*
- *a description of the tests performed and, when relevant, of the environment used, so as to allow these tests to be repeated in identical conditions;*

– *les hypothèses faites et la justification de leur validité;*

– *les résultats obtenus et la démonstration de leur validité;*

– *les conclusions atteintes et les décisions prises.*

6.2.2.3 Retours d'expérience

Les retours d'expérience peuvent être utilisés pour compléter la démonstration de conformité dans les conditions suivantes:

1 *Les retours d'expérience pris en compte doivent correspondre à des versions précisément identifiées du logiciel pré-développé, et lorsque ce logiciel est spécifique à un équipement, à des versions précisément identifiées de l'équipement.*

2 *Quand tout ou partie des retours d'expérience correspondent à d'autres versions du logiciel pré-développé et/ou de l'équipement, les différences avec les versions devant être utilisées dans le système d'I&C doivent être analysées, et l'applicabilité des retours d'expérience doit être établie.*

3 *Il doit être établi que les retours d'expérience pris en considération correspondent à des conditions couvrant celles au sein du système d'I&C ou des conditions plus sévères.*

4 *Le volume des retours d'expérience pris en considération doit être documenté.*

5 *Les méthodes utilisées pour la collecte des informations de retour d'expérience prises en considération doivent être documentées. En particulier, cette documentation doit montrer que les défaillances éventuelles causées par le logiciel pré-développé durant les retours d'expérience pris en considération ont bien été détectées et signalées.*

6 *Il doit être établi que ces défaillances ont été correctement analysées et que les défauts logiciels correspondants ont été corrigés.*

Les retours d'expérience dans des systèmes de classe de sûreté inférieure ou dans des systèmes non classés de sûreté peuvent être pris en considération pourvu que les exigences de ce paragraphe soient satisfaites.

6.2.2.4 Certifications

Un logiciel pré-développé utilisé dans des systèmes importants pour la sûreté déjà en service (mais non nécessairement dans des systèmes d'I&C de centrales nucléaires) peut avoir été certifié conforme à des normes de sûreté. Les démonstrations apportées par de telles certifications peuvent être prises en considération dans les conditions suivantes:

1 *L'identification précise du logiciel certifié doit être documentée. Si ce logiciel a été certifié dans le cadre d'un produit plus large (par exemple dans le cadre de la certification d'un équipement ou d'une famille d'équipements), l'identification précise de ce produit doit aussi être documentée.*

2 *Les démonstrations supportant la certification doivent pouvoir être évaluées, en particulier:*

– *les conditions de la certification (par exemple les conditions d'utilisation et les hypothèses faites);*

– *les méthodes et les outils utilisés pour la certification;*

– *les résultats obtenus (par exemple les propriétés et/ou les mesures certifiées).*

3 *La pertinence de ces conditions et de ces résultats pour la conformité du logiciel et pour le système d'I&C doit être établie.*

4 *Il convient que l'efficacité des méthodes et des outils utilisés pour la certification soit établie.*

5 *L'entité certificatrice doit être identifiée et doit être compétente pour les propriétés et/ou les mesures certifiées.*

6 *La version du logiciel certifié doit être la même que celle qui est utilisée dans le système d'I&C.*

- *the hypotheses made, and evidence of their validity;*
- *the results obtained, and evidence of their correctness;*
- *the conclusions reached and the resolutions agreed.*

6.2.2.3 Operational experience

Operational experience may be used to complement the justification of correctness of a pre-developed software, under the following conditions:

1 *The operational experience taken into consideration shall correspond to precisely identified versions of the pre-developed software and, when this software is specific to equipment, of the equipment in which it operates.*

2 *When all or part of the operational experience corresponds to other versions of the pre-developed software and/or of the equipment, the differences with the versions to be used in the I&C system shall be assessed, and the relevance of this operational experience shall be justified.*

3 *Documented justification shall be given that the operational experience taken into consideration corresponds to conditions of use covering those in the I&C system, or were even more severe.*

4 *The volume of the operational experience taken into consideration shall be documented.*

5 *The methods used for collecting the operational experience taken into consideration shall be documented. In particular, documented justification shall be given that the failures (if any) caused by the pre-developed software during the operational experience taken into consideration were correctly detected and reported.*

6 *Evidence shall be provided that these failures were correctly analysed, and that the corresponding software faults corrected.*

Operational experience in systems of a lower safety class, or in non-safety classified systems may be taken into consideration, provided that the requirements of this Subclause are satisfied.

6.2.2.4 Certification

Pre-developed software used in systems important to safety already in operation (albeit not necessarily in I&C systems of nuclear power plants) may have been certified for compliance to some safety standards. The evidence provided by such a certification may be taken into consideration under the following conditions:

1 *The precise identification of the pre-developed software certified shall be documented. If it was certified as a part of a larger product (for example, as a part of an equipment or equipment family), the precise identification of this product shall also be documented.*

2 *The evidence supporting the certification shall be assessable, in particular:*

- *the conditions (for example, the conditions of use and the assumptions) of the certification;*
- *the methods and tools used for the certification;*
- *the results obtained (for example, the properties and/or measurements certified).*

3 *The relevance of these conditions and results to correctness and to the I&C system shall be justified.*

4 *The effectiveness of the methods and tools used for the certification should be justified.*

5 *The certifying authority shall be identified and shall be competent for the properties and/or measurements certified.*

6 *The version of the pre-developed software certified shall be the same as the one used in the I&C system.*

6.2.2.5 Modification

Quand une modification limitée et clairement identifiée est faite dans un logiciel pré-développé pour lequel une démonstration appropriée de conformité existe déjà, ou lorsqu'il est nécessaire de supprimer des défauts, les exigences du présent paragraphe peuvent se substituer aux exigences de 6.2.2.1 à 6.2.2.4 pour mettre à jour ou compléter la démonstration. Une altération de la configuration du logiciel pré-développé ne constitue pas une modification si la nouvelle configuration reste dans le domaine couvert par la démonstration.

1 *La modification d'un logiciel pré-développé doit être documentée. Cette documentation doit préciser:*

   – *l'identité précise du logiciel modifié;*

   – *le contexte de la modification, si le logiciel fait partie d'un produit plus large (par exemple s'il fait partie d'un équipement ou d'une famille d'équipements);*

   – *les objectifs, la spécification et les contraintes de la modification;*

   – *les changements apportés à la Documentation pour la Sûreté.*

   Il convient aussi qu'elle précise les changements apportés à la conception du logiciel pré-développé.

Le contexte d'une modification peut par exemple indiquer:

- *l'identité précise du produit modifié;*
- *les objectifs, la spécification et les contraintes de la modification du produit;*
- *les modifications dans le reste du produit qui doivent être faites ou qui peuvent avoir un impact sur le logiciel pré-développé;*
- *les actions de vérification et de validation réalisées au niveau du produit.*

2 *Une démonstration documentée (par exemple basée sur des inspections manuelles, sur des analyses outillées et/ou sur des tests) concernant le logiciel modifié et éventuellement le produit qui le contient doit établir que:*

   – *les objectifs de la modification sont satisfaits;*

   – *aucun défaut n'a été introduit;*

   – *le logiciel modifié est conforme à sa Documentation pour la Sûreté mise à jour.*

3 *Le caractère suffisant de cette démonstration doit être établi, éventuellement en considérant les modifications effectuées et les conditions d'utilisation au sein du système d'I&C.*

6.2.3 Adéquation fonctionnelle

L'objectif de ce paragraphe est de s'assurer qu'un logiciel pré-développé répond bien aux besoins du système d'I&C et qu'il n'est pas trop complexe au regard de ces besoins.

1 *Le cas échéant, la Documentation pour la Sûreté d'un logiciel pré-développé doit être évaluée en regard des Spécifications du Système et de la Conception du Système. Les incohérences doivent être résolues.*

2 *Il convient que les fonctions du logiciel pré-développé non nécessaires à la satisfaction de la Spécification du Système soient identifiées. Il convient que leur innocuité soit établie.*

6.2.4 Sélection et utilisation d'équipements contenant du logiciel

Des équipements en boîte noire contenant du logiciel peuvent être utilisés aux conditions suivantes:

1 Le logiciel de l'équipement doit lui être intégré de façon à ne pas pouvoir être modifié par l'utilisateur et à ne pas pouvoir être utilisé séparément du reste de l'équipement.

6.2.2.5 Modification

When a well-identified and limited modification is made on pre-developed software for which an appropriate justification of correctness already exists, or when it is necessary to remove faults, the following requirements can be used as a substitute for the requirements of 6.2.2.1 to 6.2.2.4 to update or complete the justification. A change in the configuration of the pre-developed software does not constitute a modification, provided that the new configuration remains within the range covered by the justification.

1 *The modification of pre-developed software shall be documented. The documentation shall state:*

- *the precise identification of the modified software;*
- *the context of the modification, if the software is a part of a larger product (for example, an equipment or equipment family);*
- *the objectives, the specification and the constraints of the modification;*
- *the changes made to the Documentation for Safety.*

It should also state the changes made to the design of the pre-developed software.

The context of a modification may for example indicate:

- *the precise identification of the modified larger product;*
- *the objectives, the specification and the constraints of the modification of the product;*
- *the modifications in the rest of the product that need to be made or that may have an impact on the pre-developed software;*
- *the verification and validation actions performed at the level of the product.*

2 *Documented evidence (for example, based on manual inspections, tool supported analyses and/or tests) regarding the modified software, and possibly the larger product, shall justify that:*

- *the objectives of the modification are satisfied;*
- *no faults have been introduced;*
- *the modified software conforms to its updated Documentation for Safety.*

3 *The sufficiency of this evidence shall be justified, possibly taking into account the modifications made and the conditions of use within the I&C system.*

6.2.3 Functional suitability

The objective of this Subclause is to ensure that a pre-developed software is well-suited for the needs of the I&C system, and that it is not too complex with respect to these needs.

1 *When applicable, the Documentation for Safety of pre-developed software shall be evaluated with respect to System Specification and System Design. Inconsistencies shall be resolved.*

2 *The functions of the pre-developed software which are not required to support the System Requirements Specifications should be identified. Justification of harmlessness should be given.*

6.2.4 Selection and use of dedicated devices with embedded software

Black-box devices with embedded software may be used in an I&C system under the following conditions.

1 The software of the device shall be integrated in such a way that it cannot be modified by the user and cannot be used separately from the rest of the device.

2 Le nombre des fonctions de l'équipement, ses possibilités de configuration et l'étendue de ses interfaces et de ses interactions avec le reste du système d'I&C doivent permettre une bonne couverture fonctionnelle par des tests.

3 Il doit être établi que l'équipement est conforme aux exigences de 6.2.1, 6.2.2 et 6.2.3.

4 Il doit être établi que la configuration et l'utilisation de l'équipement dans le système d'I&C sont conformes aux exigences de 6.4, 6.5.1 et 6.5.2.

5 La conception du système d'I&C et de son logiciel doivent garantir que l'équipement est utilisé dans des conditions clairement définies.

6.3 Spécification du logiciel

Ce paragraphe complète et précise les exigences de 6.1.2.3 de la CEI 61513.

6.3.1 Objectifs

1 Les exigences sur le logiciel d'un système d'I&C doivent être spécifiées et documentées.

Dans cette Norme, le document ou l'ensemble de documents correspondant est appelé la Spécification du Logiciel. En principe, son objectif est de préciser ce que le logiciel doit accomplir en évitant de spécifier comment le réaliser. Cependant, des contraintes de conception et de réalisation peuvent être spécifiées si elles sont nécessaires compte tenu de la conception du système d'I&C ou de l'architecture d'I&C.

2 Il convient que la Spécification du Logiciel évite de compliquer inutilement la conception du logiciel. Il convient également qu'elle vise à offrir au logiciel des conditions d'utilisation stables.

3 La Spécification du Logiciel doit être telle:

- qu'elle contribue à l'établissement que la conception du système d'I&C est correcte;
- que la satisfaction des exigences de la CEI 61513 par le système d'I&C puisse être démontrée.

Les exigences de la CEI 61513 concernées par la Spécification du Logiciel sont principalement en 6.1.1.2, 6.1.1.3, 6.1.1.4, 6.1.2.2, 6.1.2.4 et 6.1.3.

4 La Spécification du Logiciel doit être une référence pour la conception et la validation du logiciel, ainsi que pour les modifications éventuelles.

6.3.2 Entrées

1 Les entrées de la Spécification du Logiciel doivent inclure la Spécification du Système et la Documentation de Conception du Système.

Il peut aussi y avoir d'autres entrées, comme les contraintes spécifiques du projet et/ou les règles et normes applicables.

2 Il convient que la structure de la Spécification du Logiciel facilite la vérification de sa conformité et de son exhaustivité par rapport à ses entrées.

La Spécification du Logiciel peut faire référence à des parties de ses entrées de façon à éviter des duplications inutiles et à minimiser les risques d'incohérence. Elle peut aussi faire référence à des documents déjà existants, tels que la documentation des logiciels pré-développés.

3 Les références éventuelles faites par la Spécification du Logiciel à d'autres documents doivent être précises de façon à éviter toute ambiguïté.

4 Il convient que la Spécification du Logiciel évite les ajouts inutiles par rapport à ses entrées.

2 The complexity of the functions of the device, its potential for configuration, and the extent of its interfaces and interactions with the rest of the I&C system shall be limited so as to allow a thorough functional coverage by tests.

3 Evidence shall be given that the device conforms to the requirements of 6.2.1, 6.2.2 and 6.2.3.

4 Evidence shall be given that the configuration and the use of the device in the I&C system conforms to the requirements of 6.4, 6.5.1 and 6.5.2.

5 The design of the I&C system and/or of its software shall ensure that the device is used in clearly defined conditions.

6.3 Software requirements specification

This Subclause completes and adds precision to the requirements of 6.1.2.3 of IEC 61513.

6.3.1 Objectives

1 The requirements for the software of an I&C system shall be specified and documented.

In this standard, the corresponding document or set of documents is called the Software Requirements Specification. In principle, its objective is to specify what the software is to achieve without specifying how it must do it. However, design and implementation constraints may have to be specified when this is required by considerations of the design of the I&C system or of the I&C architecture.

2 The Software Requirements Specification should avoid unnecessary complexity of the software design, and should aim at providing stable conditions of use for the software.

3 The Software Requirements Specification shall be such that:

- it contributes to the confidence in the correctness of the design of the I&C system;
- compliance of the I&C system to the requirements of IEC 61513 can be demonstrated.

The IEC 61513 requirements concerned with Software Requirements Specification are mainly in Subclauses 6.1.1.2, 6.1.1.3, 6.1.1.4, 6.1.2.2, 6.1.2.4 and 6.1.3.

4 The Software Requirements Specification shall be a reference for software design, software validation, and possible software modifications.

6.3.2 Inputs

1 The inputs to the Software Requirements Specification shall include the System Specification and the System Design Documentation.

They may also include other documents, such as project specific constraints, and/or applicable rules and standards.

2 The structure of the Software Requirements Specification should facilitate verification to ensure that it is consistent and complete with respect to its input documents.

The Software Requirements Specification may reference parts of its inputs, so as to avoid unnecessary duplications and minimise the risk of inconsistencies. It may also reference other pre-existing documents, such as the documentation of pre-developed software.

3 The references, if any, made by the Software Requirements Specification to other documents shall be precise so as to be unambiguous.

4 The Software Requirements Specification should avoid unnecessary additions with respect to its inputs.

En principe, il est préférable que le logiciel n'ait pas plus de fonctionnalités que ce qui est exigé afin de minimiser la complexité. Cependant, les pratiques industrielles actuelles étant basées sur l'utilisation de composants pré-développés, l'introduction de capacités non requises peut être justifiée.

6.3.3 Contenu

1 La Spécification du Logiciel doit spécifier:

- les fonctions d'application devant être assurées par le logiciel;
- les différents modes de fonctionnement du logiciel, ainsi que les conditions de transition correspondantes;
- les interfaces et les interactions du logiciel avec son environnement (par exemple avec les opérateurs, avec le reste du système d'I&C, et avec les autres systèmes et équipements avec lesquels il interagit ou partage des ressources), et en particulier les rôles, types, formats, domaines de valeur et contraintes des entrées et des sorties;
- les paramètres du logiciel pouvant être modifiés par les opérateurs en cours d'exploitation, s'il y a lieu, ainsi que leurs rôles, types, formats, domaines de valeur et contraintes, et les contrôles devant être réalisés par le logiciel en cas de modification;
- les performances requises, lorsque cela est pertinent;
- ce que le logiciel ne doit pas faire ou doit éviter, lorsque cela est pertinent;
- les attentes ou les suppositions du logiciel sur son environnement, s'il y a lieu.

2 Il convient que la Spécification du Logiciel spécifie également les conditions que l'environnement offre au logiciel (par exemple les taux de sollicitation), et en particulier les conditions extrêmes.

Les exigences de fonctionnalité, d'interface et de performance peuvent dépendre du mode de fonctionnement, des valeurs des paramètres, des données de configuration, et des conditions offertes au logiciel.

3 La Spécification du Logiciel doit spécifier les modes de fonctionnement du logiciel en cas de détection d'erreur ou de défaillance. Lorsque des tests périodiques sont exigés du système d'I&C, la Spécification du Logiciel doit aussi spécifier le mode de fonctionnement à adopter au cours de ces tests.

4 Il convient que la Spécification du Logiciel précise les objectifs de qualité pour le logiciel. Elle doit préciser les contraintes devant être respectées pour que la conception et la réalisation du logiciel soient correctes et robustes.

Ceci peut inclure des contraintes visant:

- à garantir que le logiciel et la conception du système sont corrects (par exemple des marges dans la gestion des ressources allouées dynamiquement comme la mémoire, la puissance de traitement, la bande passante des canaux de communication et les ressources du système d'exploitation);
- à augmenter la capacité du logiciel et du système d'I&C à tolérer les défauts, à détecter et signaler les erreurs et défaillances, à adopter les modes de fonctionnement spécifiés et à récupérer après une défaillance;
- à garantir que les erreurs des opérateurs et les défaillances des autres systèmes et équipements avec lesquels le logiciel interagit ou partage des ressources n'auront pas de conséquences inacceptables.

5 La Spécification du Logiciel doit spécifier la contribution du logiciel à l'assurance que les opérateurs seront informés en temps voulu des erreurs et défaillances concernant les fonctions du système d'I&C identifiées comme importantes pour la sûreté. Les informations délivrées aux opérateurs doivent leur permettre d'entreprendre toute action appropriée.