NF EN 62304

OCTOBRE 2006

www.afnor.org

Contacter :
AFNOR – Norm'Info
11, rue Francis de Pressensé
93571 La Plaine Saint-Denis Cedex
Tél : 01 41 62 76 44
Fax : 01 49 17 92 02
E-mail : norminfo@afnor.org

afnor

norme européenne

norme française

NF EN 62304
Octobre 2006

Indice de classement : **C 74-017**

ICS : 11.040

Logiciels de dispositifs médicaux
Processus du cycle de vie du logiciel

E : Medical device software - Software life-cycle processes

D : Medizingeräte-Software - Software-Lebenszyklus-Prozesse

Norme française homologuée

par décision du Directeur Général d'AFNOR le 5 septembre 2006 pour prendre effet à compter du 5 octobre 2006.

Correspondance

La norme européenne EN 62304:2006 + corrigendum de novembre 2008 a le statut d'une norme française. Elle reproduit intégralement la publication CEI 62304:2006.

Analyse

Le présent document définit les exigences du cycle de vie des LOGICIELS DE DISPOSITIFS MEDICAUX. L'ensemble des PROCESSUS, ACTIVITES et TACHES décrit dans le présent document constitue un cadre commun pour les PROCESSUS du cycle de vie des LOGICIELS DE DISPOSITIFS MEDICAUX.

dow : 2009-06-01

Le présent document entre dans le champ d'application de la Directive n°93/42/CEE du 14/06/1993 relative aux dispositifs médicaux, de la Directive n°90/385/CEE du 20/06/1990, concernant le rapprochement des législations des États membres relatives aux dispositifs médicaux implantables actifs et de la Directive n°98/79/CE du 27/10/1998 relative aux dispositifs médicaux de diagnostic in-vitro.

Descripteurs

Appareil électromédical, dispositif médical, logiciel, cycle de vie, processus, maintenance, qualité, gestion, risque.

Modifications

Corrections

Par rapport au premier tirage, modification de l'avant-propos et ajout de l'annexe ZZ.

éditée et diffusée par l'Union Technique de l'Electricité (UTE) – Tour Chantecoq – 5, rue Chantecoq – 92808 Puteaux Cedex – Tél. : + 33 (0) 1 49 07 62 00 – Télécopie : + 33 (0) 1 47 78 73 51 – Courriel : ute@ute.asso.fr – Internet : http://www.ute-fr.com/
diffusée également par l'Association Française de Normalisation (AFNOR) – 11, avenue Francis de Pressensé – 93571 Saint-Denis La Plaine Cedex – Tél. : + 33 (0) 1 41 62 80 00

Impr. UTE

2ème tirage 2009-01 – Reproduction interdite

AVANT-PROPOS NATIONAL

Ce document constitue la version française complète de la norme européenne EN 62304:2006 en reprenant le texte de la publication CEI 62304:2006.

Les modifications du CENELEC (dans le présent document l'annexe ZA uniquement) sont signalées par un trait vertical dans la marge gauche du texte.

L'Union Technique de l'Électricité a voté favorablement au CENELEC sur le projet de EN 62304, le 13 avril 2006.

———————

Correspondance entre les documents internationaux cités en référence et les documents CENELEC et/ou français à appliquer

Document international cité en référence	**Document correspondant**	
	CENELEC (EN ou HD)	**français** (NF ou UTE)
ISO 14971	EN ISO 14971 (2000)	NF EN 14971 (2001) (S 99-211)

Note : Les documents de la classe C sont en vente à l'Union Technique de l'Électricité – Tour Chantecoq – 5, rue Chantecoq – 92808 Puteaux Cedex – Tél. : 01 49 07 62 00 – ainsi qu'au service diffusion de l'Association française de normalisation – 11, avenue Francis de Pressensé – 93571 Saint-Denis La Plaine Cedex – Tél. : 01 41 62 80 00.

Les documents CEI sont en vente à l'UTE.

Les documents ISO et de la classe S sont en vente à AFNOR.

———————

NORME EUROPÉENNE

EUROPÄISCHE NORM

EUROPEAN STANDARD

EN 62304

Juillet 2006

ICS 11.040

Version française

Logiciels de dispositifs médicaux
Processus du cycle de vie du logiciel
(CEI 62304:2006)

Medizingeräte-Software -
Software-Lebenszyklus-Prozesse
(IEC 62304:2006)

Medical device software -
Software life-cycle processes
(IEC 62304:2006)

La présente Norme Européenne a été adoptée par le CENELEC le 2006-06-01. Les membres du CENELEC sont tenus de se soumettre au Règlement Intérieur du CEN/CENELEC qui définit les conditions dans lesquelles doit être attribué, sans modification, le statut de norme nationale à la Norme Européenne.

Les listes mises à jour et les références bibliographiques relatives à ces normes nationales peuvent être obtenues auprès du Secrétariat Central ou auprès des membres du CENELEC.

La présente Norme Européenne existe en trois versions officielles (allemand, anglais, français). Une version dans une autre langue faite par traduction sous la responsabilité d'un membre du CENELEC dans sa langue nationale, et notifiée au Secrétariat Central, a le même statut que les versions officielles.

Les membres du CENELEC sont les comités électrotechniques nationaux des pays suivants: Allemagne, Autriche, Belgique, Chypre, Danemark, Espagne, Estonie, Finlande, France, Grèce, Hongrie, Irlande, Islande, Italie, Lettonie, Lituanie, Luxembourg, Malte, Norvège, Pays-Bas, Pologne, Portugal, République Tchèque, Roumanie, Royaume-Uni, Slovaquie, Slovénie, Suède et Suisse.

CENELEC

Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung
European Committee for Electrotechnical Standardization

Secrétariat Central: rue de Stassart 35, B - 1050 Bruxelles

Ref. n° EN 62304:2006 F

Avant-propos

Le texte du document 62A/523/FDIS, future édition 1 de la CEI 62304, préparé par un groupe de travail mixte du SC 62A, Aspects généraux des équipements utilisés en pratique médicale, du comité d'études 62 de la CEI, Equipements électriques dans la pratique médicale, et du comité technique 210 de l'ISO, Management de la qualité et aspects généraux correspondants des dispositifs médicaux, a été soumis au vote parallèle CEI-CENELEC et a été approuvé par le CENELEC comme EN 62304 le 2006-06-01.

Les dates suivantes ont été fixées:

– date limite à laquelle la EN doit être mise en application au niveau national par publication d'une norme nationale identique ou par entérinement	(dop)	2007-03-01
– date limite à laquelle les normes nationales conflictuelles doivent être annulées	(dow)	2009-06-01

Les polices de caractère suivantes sont utilisées dans la présente norme:

– exigences et définitions: en caractères romains;

– des éléments d'information apparaissant hors des tableaux tels que les notes, les exemples et les références: en petits caractères. Le texte normatif des tableaux est également en petits caractères;

– les termes utilisés partout dans la présente norme, qui ont été définis dans l'Article 3 et énumérés également dans l'index: EN PETITES MAJUSCULES.

Lorsqu'un astérisque (*) est utilisé comme premier caractère d'un titre ou au début d'un paragraphe, il indique que des lignes directrices relatives à cet élément sont fournies en Annexe B.

Le Tableau C.5 a été préparé par le Comité Technique mixte ISO/CEI 1/SC7, Ingénierie du logiciel et du système.

La présente Norme Européenne a été élaborée dans le cadre d'un mandat donné au CENELEC par la Commission des Communautés Européennes et l'Association Européenne de Libre Echange et couvre les exigences essentielles des Directives 93/42/CEE, 90/385/CE et 98/79/CE. Voir l'Annexe ZZ.

Les Annexes ZA et ZZ ont été ajoutées par le CENELEC.

_____

SOMMAIRE

INTRODUCTION

Le logiciel fait souvent partie intégrante de la technologie des DISPOSITIFS MEDICAUX. La détermination de la SECURITE et de l'efficacité d'un DISPOSITIF MEDICAL comportant un logiciel exige que soit connu ce qu'il est prévu que le logiciel accomplisse et qu'il soit démontré que son utilisation remplit ces objectifs sans entraîner de RISQUES inacceptables.

La présente norme fournit un cadre pour les PROCESSUS du cycle de vie en définissant les ACTIVITES et TACHES nécessaires à la conception et à la maintenance en toute SECURITE des LOGICIELS DE DISPOSITIFS MEDICAUX. La présente norme fournit les exigences applicables à chaque PROCESSUS du cycle de vie. Chaque PROCESSUS du cycle de vie est en outre divisé en un ensemble D'ACTIVITES dont la plupart sont ensuite divisées en un ensemble de TACHES.

On suppose par principe que les LOGICIELS DE DISPOSITIFS MEDICAUX sont développés et maintenus dans le cadre d'un système de management de la qualité (voir 4.1) et d'un système de GESTION DES RISQUES (voir 4.2). Le PROCESSUS DE GESTION DES RISQUES est déjà parfaitement traité dans la Norme Internationale ISO 14971. En conséquence, la norme CEI 62304 tire profit de cet avantage par une simple référence normative à l'ISO 14971. Cependant, pour les logiciels, des exigences supplémentaires mineures de GESTION DES RISQUES sont nécessaires, notamment dans le domaine de l'identification des facteurs contributifs des logiciels en termes de DANGER. Ces exigences sont résumées et introduites dans l'Article 7, PROCESSUS DE GESTION DES RISQUES liés au logiciel.

L'éventuelle contribution d'un logiciel à un DANGER donné est déterminée lors de L'ACTIVITE d'identification des DANGERS du PROCESSUS DE GESTION DES RISQUES. LES DANGERS qui pourraient être indirectement induits par les logiciels (par exemple la fourniture d'informations propres à induire en erreur qui pourrait donner lieu à l'administration d'un traitement inadéquat) doivent être pris en compte lorsqu'il s'agit de déterminer si le logiciel est un facteur contributif. La décision d'utiliser le logiciel pour maîtriser les RISQUES est prise lors de L'ACTIVITE DE MAITRISE DES RISQUES du PROCESSUS DE GESTION DES RISQUES. Le PROCESSUS DE GESTION DES RISQUES lié au logiciel prescrit dans la présente norme doit être intégré au PROCESSUS DE GESTION DES RISQUES lié au dispositif conformément à l'ISO 14971.

Le PROCESSUS de développement des logiciels couvre un certain nombre d'ACTIVITES. Ces ACTIVITES sont illustrées en Figure 1 et décrites dans l'Article 5. Parce qu'il est notoire que de nombreux incidents sur le terrain sont liés à l'entretien ou à la maintenance des SYSTEMES DE DISPOSITIFS MEDICAUX comprenant des mises à jour et des mises à niveau inadéquates du logiciel, on considère que le PROCESSUS de maintenance des logiciels est aussi important que le PROCESSUS de développement des logiciels. Le PROCESSUS de maintenance des logiciels est très similaire au PROCESSUS de développement des logiciels. Cela est illustré en Figure 2 et décrit dans l'Article 6.

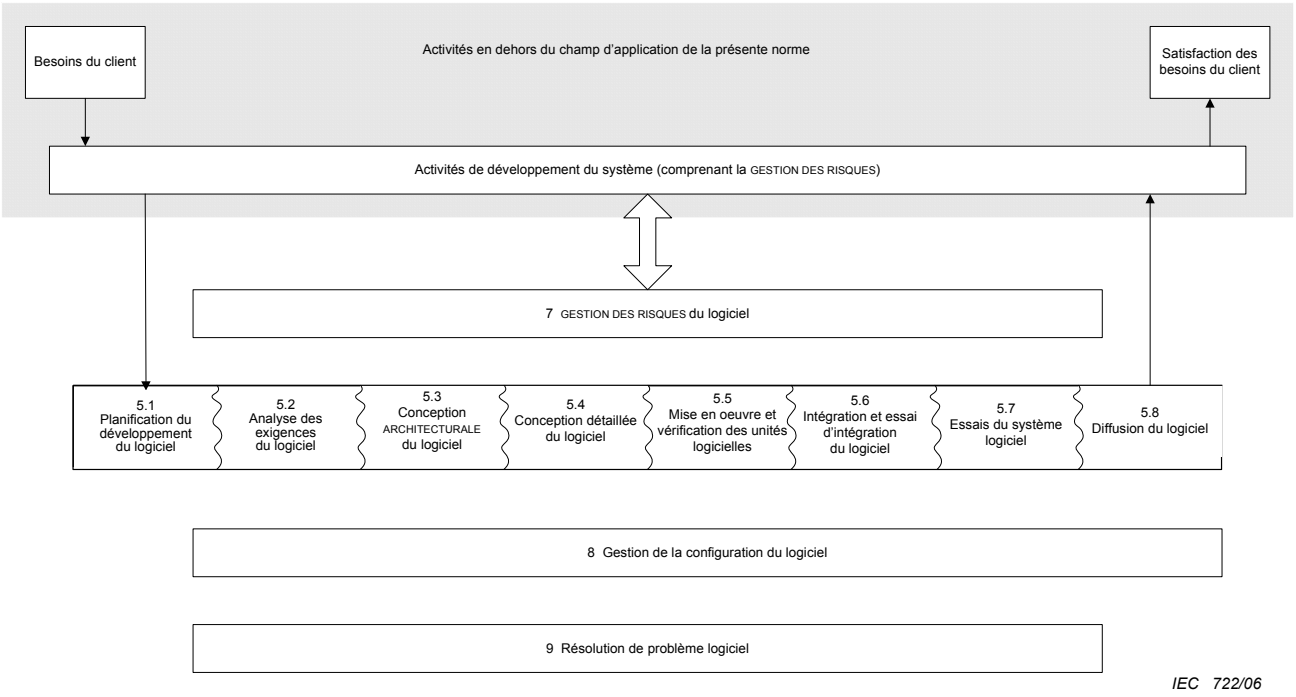

Figure 1 – Présentation générale des PROCESSUS et ACTIVITES de développement de logiciels

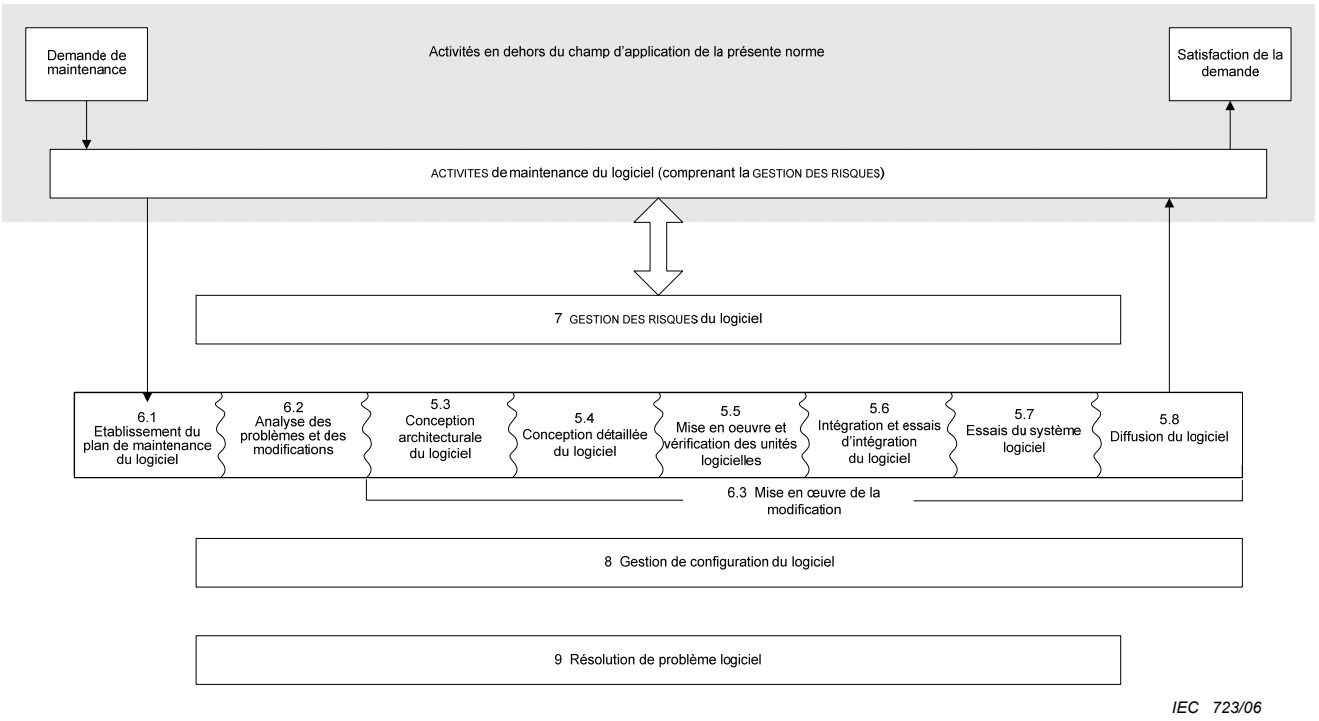

Figure 2 – Présentation générale des PROCESSUS et ACTIVITES de maintenance de logiciels

La présente norme identifie deux PROCESSUS additionnels considérés comme essentiels pour le développement de LOGICIELS DE DISPOSITIFS MEDICAUX sûrs. Il s'agit du PROCESSUS de gestion de la configuration du logiciel (Article 8) et du PROCESSUS de résolution des problèmes de logiciel (Article 9).

La présente norme ne prescrit aucune structure organisationnelle pour le FABRICANT et n'entend pas spécifier quelle organisation doit réaliser tel ou tel PROCESSUS, ACTIVITE ou TACHE. La présente norme exige uniquement que le PROCESSUS, l'ACTIVITE ou la TACHE soit mené à bien pour assurer la conformité à la présente norme.

La présente norme ne prescrit pas de désignation, de format ou de contenu explicite de la documentation à produire. Elle exige que les TACHES soient documentées, mais c'est à l'utilisateur de décider de la manière dont la documentation correspondante doit être présentée.

La présente norme ne prescrit pas un modèle de cycle de vie spécifique. Il incombe aux utilisateurs de la présente norme de choisir un modèle de cycle de vie pour un projet de logiciel et de faire correspondre les PROCESSUS, ACTIVITES et TACHES définis dans la présente norme avec ce modèle.

L'Annexe A fournit une justification des articles de la présente norme. L'Annexe B donne des conseils relatifs aux dispositions de la présente norme.

Pour les besoins de la présente norme:

- «doit» signifie qu'une exigence donnée est obligatoire pour être conforme à la présente norme;
- «il convient de – est recommandé» signifie qu'une exigence donnée est recommandée mais n'est pas obligatoire pour être conforme à la présente norme;
- «peut – est admis» est utilisé pour décrire une manière autorisée pour être conforme à une prescription donnée;
- «établir» signifie définir, documenter et mettre en œuvre; et
- Lorsque la présente norme utilise l'expression «si nécessaire» ou «le cas échéant», conjointement à un PROCESSUS, une ACTIVITE, une TACHE ou un résultat exigés, cela signifie que le FABRICANT doit utiliser le PROCESSUS, l'ACTIVITE, la TACHE ou le résultat et dans le cas contraire il doit justifier sa décision par écrit.

LOGICIELS DE DISPOSITIFS MEDICAUX
PROCESSUS DU CYCLE DE VIE DU LOGICIEL

1 Domaine d'application

1.1 *Objet

La présente norme définit les exigences du cycle de vie des LOGICIELS DE DISPOSITIFS MEDICAUX. L'ensemble des PROCESSUS, ACTIVITES et TACHES décrit dans la présente norme constitue un cadre commun pour les PROCESSUS du cycle de vie des LOGICIELS DE DISPOSITIFS MEDICAUX.

1.2 * Domaine d'application

La présente norme s'applique au développement et à la maintenance des LOGICIELS DE DISPOSITIFS MEDICAUX.

La présente norme s'applique au développement et à la maintenance des LOGICIELS DE DISPOSITIFS MEDICAUX lorsque le logiciel est un DISPOSITIF MEDICAL ou lorsque le logiciel est incorporé ou fait partie intégrante du DISPOSITIF MEDICAL final.

La présente norme ne couvre pas la validation et la mise sur le marché du DISPOSITIF MEDICAL, même lorsque le DISPOSITIF MEDICAL est intégralement constitué du logiciel.

1.3 Relations avec d'autres normes

La présente norme couvrant le cycle de vie des LOGICIELS DE DISPOSITIFS MEDICAUX doit être utilisée conjointement à d'autres normes pertinentes pour le développement d'un DISPOSITIF MEDICAL. L'Annexe C présente les relations existant entre la présente norme et d'autres normes pertinentes.

1.4 Conformité

La conformité à la présente norme est définie comme la mise en œuvre de tous les PROCESSUS, ACTIVITES et TACHES identifiés dans la présente norme en fonction de la classe de sécurité.

NOTE Les classes de sécurité du logiciel assignées à chaque exigence sont identifiées dans le texte normatif suivant l'exigence.

La conformité est déterminée par inspection de toute documentation exigée par la présente norme y compris le DOSSIER DE GESTION DES RISQUES et l'évaluation des PROCESSUS, ACTIVITES et TACHES requis pour la classe de SECURITE du logiciel. Voir l'Annexe D.

NOTE 1 Cette évaluation peut être effectuée par audit interne ou externe.

NOTE 2 Même lorsque les PROCESSUS, ACTIVITES et TACHES sont effectivement réalisés, il existe une certaine flexibilité dans les méthodes de mise en œuvre de ces PROCESSUS et d'exécution de ces ACTIVITES et TACHES.

NOTE 3 Lorsqu'une éventuelle exigence comporte la mention «le cas échéant» ou «si nécessaire» et qu'elle n'est pas réalisée, une justification écrite est nécessaire pour cette évaluation.

NOTE 4 Dans la version anglaise de l'ISO/CEI 12207 le terme «conformance» est utilisé pour «conformité», alors que dans la version anglaise de la présente norme, on utilise le terme «compliance».

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).

ISO 14971, *Dispositifs médicaux – Application de la gestion des risques aux dispositifs médicaux*.

3 *Termes et définitions

Pour les besoins du présent document, les termes et les définitions qui suivent s'appliquent.

3.1
ACTIVITE
ensemble d'une ou de plusieurs TACHES corrélées ou interactives

3.2
ANOMALIE
tout état qui s'écarte de ce qui est attendu sur la base des spécifications des exigences, des documents de conception, des normes, etc., ou qui ne correspond pas à la perception ou à l'expérience d'un individu donné. Les ANOMALIES peuvent être décelées, sans limitation aucune, pendant la revue, l'essai, l'analyse, la compilation ou l'utilisation des PRODUITS LOGICIELS ou de la documentation applicable

[IEEE 1044:1993, définition 3.1]

3.3
ARCHITECTURE
structure organisationnelle d'un SYSTEME ou d'un composant

[IEEE 610.12:1990]

3.4
DEMANDE DE MODIFICATION
spécification écrite d'une modification à effectuer sur un PRODUIT LOGICIEL

3.5
ELEMENT DE CONFIGURATION
entité qui peut être identifiée de manière univoque en un point de référence donné

NOTE Basé sur l'ISO/CEI 12207:1995, définition 3.6

3.6
LIVRABLE
résultat ou élément de sortie requis (y compris la documentation) d'une ACTIVITE ou d'une TACHE

3.7
EVALUATION
détermination systématique de l'étendue à laquelle l'entité répond aux critères spécifiés

[ISO/CEI 12207:1995, définition 3.9]

3.8
DOMMAGE
blessure physique ou atteinte à la santé des personnes ou atteinte aux biens ou à l'environnement

[ISO/CEI Guide 51:1999, définition 3.3]

3.9
PHENOMENE DANGEREUX (DANGER)
source potentielle de DOMMAGE

[ISO/CEI Guide 51:1999, définition 3.5]

3.10
FABRICANT
personne physique ou morale chargée de la conception, de la fabrication, du conditionnement ou de l'étiquetage d'un DISPOSITIF MEDICAL de l'assemblage d'un SYSTEME ou de l'adaptation d'un DISPOSITIF MEDICAL avant mise sur le marché et/ou mise en service indépendamment du fait que ces opérations soient effectuées par cette personne ou par une tierce partie pour le compte de cette personne

[ISO 14971:2000, définition 2.6]

3.11
DISPOSITIF MEDICAL
tout instrument, appareil, équipement, machine, dispositif, implant, réactif *in vitro* ou calibreur, logiciel, matériel ou autre article similaire ou associé dont le FABRICANT prévoit qu'il soit utilisé seul ou en association chez l'être humain pour la ou les fins spécifiques suivantes:

– diagnostic, prévention, contrôle, traitement ou atténuation d'une maladie,
– diagnostic, contrôle, traitement, atténuation ou compensation d'une blessure,
– étude, remplacement, modification ou entretien de l'anatomie ou d'un PROCESSUS physiologique,
– entretien (artificiel) ou maintien de la vie,
– maîtrise de la conception,
– désinfection des DISPOSITIFS MEDICAUX,
– communication d'informations à des fins médicales par un examen *in vitro* de spécimens (prélèvement) provenant du corps humain,

et dont l'action principale voulue dans ou sur le corps humain, n'est pas obtenue par des moyens pharmacologiques ou immunologiques ni par métabolisme mais dont la fonction peut être assistée par de tels moyens

NOTE 1 Cette définition a été élaborée par le Groupe de Travail d'Harmonisation Mondiale. Voir la référence bibliographique [15] (dans l'ISO 13485:2003).

[ISO 13485:2003, définition 3.7]

NOTE 2 Les définitions utilisées dans les réglementations de chaque pays peuvent présenter certaines différences.

3.12
LOGICIEL DE DISPOSITIF MEDICAL
SYSTEME LOGICIEL qui a été développé pour être incorporé dans le DISPOSITIF MEDICAL en cours de développement ou qui est destiné à être utilisé comme un DISPOSITIF MEDICAL à part entière

3.13
RAPPORT DE PROBLEME
enregistrement du comportement réel ou potentiel d'un PRODUIT LOGICIEL qu'un utilisateur ou une autre personne concernée considère être non sûr, inadéquat pour l'usage prévu ou contraire aux spécifications

NOTE 1 La présente norme n'exige pas que chaque RAPPORT DE PROBLEME donne lieu à une modification du PRODUIT LOGICIEL. Un FABRICANT peut en effet rejeter un RAPPORT DE PROBLEME en considérant qu'il s'agit d'un malentendu, d'une erreur ou d'un événement insignifiant.

NOTE 2 Un RAPPORT DE PROBLEME peut concerner un PRODUIT LOGICIEL diffusé ou encore en cours de développement.

NOTE 3 La présente norme exige que le FABRICANT suive des étapes décisionnelles supplémentaires (voir l'Article 6) pour un RAPPORT DE PROBLEME relatif à un produit diffusé afin de s'assurer que les mesures réglementaires pertinentes sont correctement identifiées et mises en œuvre.

3.14
PROCESSUS
ensemble D'ACTIVITES corrélées ou interactives qui transforme des éléments d'entrée en éléments de sortie

[ISO 9000:2000, définition 3.4.1]

NOTE Le terme «ACTIVITES» couvre l'utilisation des ressources.

3.15
ESSAI DE REGRESSION
essai exigé pour s'assurer qu'un changement d'un composant du SYSTEME n'a pas altéré la fonctionnalité, la fiabilité ou les performances et n'a pas entraîné de défauts supplémentaires

[ISO/CEI 90003:2004, définition 3.11]

3.16
RISQUE
combinaison de la probabilité d'un DOMMAGE et de sa gravité

[ISO/CEI Guide 51:1999, définition 3.2]

3.17
ANALYSE DU RISQUE
utilisation des informations disponibles pour identifier les PHENOMENES DANGEREUX (DANGERS) et estimer le RISQUE

[ISO/CEI Guide 51:1999, définition 3.10]

3.18
MAITRISE DU RISQUE
PROCESSUS au cours duquel des décisions sont prises et des RISQUEs sont réduits ou maintenus à des niveaux spécifiés

[ISO 14971:2000, définition 2.16, modifiée]

3.19
GESTION DES RISQUES
application systématique de politiques, de procédures et de pratiques de gestion aux TACHES d'analyse, d'évaluation et de maîtrise du RISQUE

[ISO 14971:2000, définition 2.18]

3.20
DOSSIER DE GESTION DES RISQUES
ensemble d'enregistrements et autres documents qui ne sont pas nécessairement contigus et qui sont produits par un PROCESSUS DE GESTION DES RISQUES

[ISO 14971:2000, définition 2.19]

3.21
SECURITE
absence de RISQUE inacceptable

[ISO/CEI Guide 51:1999, définition 3.1]

3.22
SURETE
protection des informations et des données de sorte que des personnes ou des SYSTEMES non autorisés ne puissent les lire ou les modifier et que l'accès à ces informations et données ne soit pas refusé à des personnes ou des SYSTEMES autorisés

[ISO/CEI 12207:1995, définition 3.25]

3.23
BLESSURE GRAVE
blessure ou maladie qui, directement ou indirectement:

a) menace la vie,

b) entraîne une carence permanente d'une fonction physiologique ou endommage de manière définitive une structure du corps, ou

c) nécessite une intervention médicale ou chirurgicale pour prévenir une carence permanente d'une fonction physiologique ou un endommagement définitif d'une structure du corps

NOTE Carence permanente signifie une carence irréversible ou un endommagement d'une structure du corps ou d'une fonction, à l'exclusion des carences ou préjudices insignifiants.

3.24
MODELE DU CYCLE DE VIE DE DEVELOPPEMENT DU LOGICIEL
structure conceptuelle couvrant la vie du logiciel depuis la définition de ses exigences jusqu'à sa mise en fabrication et qui:

– identifie le PROCESSUS, les ACTIVITES et TACHES impliqués dans le développement d'un PRODUIT LOGICIEL,

– décrit l'ordre et la dépendance entre ACTIVITES et TACHES, et

– identifie les repères auxquels la complétude des LIVRABLES spécifiés est vérifiée.

NOTE Basée sur la définition 3.11 de l'ISO/CEI 12207:1995

3.25
ELEMENT LOGICIEL
toute partie identifiable d'un programme informatique

[ISO/CEI 90003:2004, définition 3.14, modifiée]

NOTE Trois termes identifient la décomposition du logiciel. Le niveau supérieur est le SYSTEME LOGICIEL. Le niveau le plus bas qui n'est pas décomposé plus en avant est l'UNITE LOGICIELLE. Tous les niveaux de composition, y compris les niveaux supérieur et inférieur, peuvent être dénommés ELEMENTS LOGICIEL. Un SYSTEME LOGICIEL est donc composé d'un ou de plusieurs ELEMENTS LOGICIEL, et chaque ELEMENT LOGICIEL est composé d'une ou de plusieurs UNITES LOGICIELLES ou d'un ou de plusieurs ELEMENTS LOGICIELS décomposables. Il incombe au FABRICANT de fournir la définition et la granularité des ELEMENTS LOGICIELS et des UNITES LOGICIELLES.

3.26
PRODUIT LOGICIEL
ensemble constitué de programmes informatiques, de procédures, et des données et documentation éventuellement associées

[ISO/CEI 12207:1995, définition 3.26]

3.27
SYSTEME LOGICIEL
ensemble intégré d'ELEMENTS LOGICIELS organisé de manière à réaliser une fonction ou un ensemble de fonctions spécifiques

3.28
UNITE LOGICIELLE
ELEMENT LOGICIEL qui n'est pas subdivisé en d'autres éléments

NOTE Les UNITES LOGICIELLES peuvent être utilisées pour essais ou gestion de la configuration du logiciel.

3.29
SOUP (sigle pour l'anglais «Software Of Unknown Provenance»)
logiciel de provenance inconnue
ELEMENT LOGICIEL qui est déjà développé, et généralement disponible, et qui n'a pas été développé pour être incorporé dans le DISPOSITIF MEDICAL (également appelé «logiciel de série») ou logiciel précédemment développé pour lequel les enregistrements suffisants des processus de développement ne sont pas disponibles

3.30
SYSTEME
ensemble composite intégré constitué d'un ou de plusieurs PROCESSUS, matériels, logiciels, fonctionnalités et individus qui fournissent une aptitude à satisfaire un besoin ou un objectif déclaré

[ISO/CEI 12207:1995, définition 3.31]

3.31
TACHE
partie unique d'un travail qui doit être effectué

3.32
TRAÇABILITE
degré auquel une relation peut être établie entre deux ou plusieurs produits du PROCESSUS de développement

[IEEE 610.12:1990]

3.33
VERIFICATION
confirmation par des preuves tangibles que les exigences spécifiées ont été satisfaites

NOTE 1 Le terme «vérifié» désigne l'état correspondant.

[ISO 9000:2000, définition 3.8.4]

NOTE 2 En conception et développement, la VERIFICATION est le PROCESSUS d'examen du résultat d'une ACTIVITE donnée afin de déterminer la conformité à la prescription définie pour ladite ACTIVITE.

3.34
VERSION
instance identifiée d'un ELEMENT DE CONFIGURATION

NOTE 1 La modification d'une VERSION d'un PRODUIT LOGICIEL, donnant lieu à une nouvelle VERSION exige une action de gestion de la configuration du logiciel.

NOTE 2 Basé sur [ISO/CEI 12207:1995, définition 3.37]

4 * Exigences générales

4.1 * Système de management de la qualité

Le FABRICANT du LOGICIEL DE DISPOSITIF MEDICAL doit démontrer la capacité à fournir un LOGICIEL DE DISPOSITIF MEDICAL qui réponde de manière cohérente aux exigences du client et aux exigences réglementaires applicables.

NOTE 1 La démonstration de cette capacité peut se faire par l'utilisation d'un système de management de la qualité conforme à:

- l'ISO 13485 [7] ; ou
- une norme nationale de système de management de la qualité ; ou
- un système de management de la qualité exigé par une réglementation nationale.

NOTE 2 L'ISO/CEI 90003 [11] fournit des lignes directrices pour l'application des exigences d'un système de management de la qualité au logiciel.

4.2 * GESTION DES RISQUES

Le FABRICANT doit appliquer un PROCESSUS DE GESTION DES RISQUES conforme à l'ISO 14971.

4.3 * Classification de sécurité du logiciel

a) Le FABRICANT doit attribuer à chaque SYSTEME LOGICIEL une classe de SECURITE du logiciel (A, B ou C) en fonction des effets possibles sur le patient, l'opérateur ou d'autres personnes résultant d'un PHENOMENE DANGEREUX auquel le SYSTEME LOGICIEL peut contribuer.

Les classes de SECURITE du logiciel doivent au départ être attribuées en se basant sur le degré de sévérité suivant:

Classe A : Aucune blessure ou atteinte à la santé n'est possible

Classe B : Une BLESSURE NON GRAVE est possible

Classe C : La mort ou une BLESSURE GRAVE est possible

Si le PHENOMENE DANGEREUX peut résulter d'une défaillance du SYSTEME LOGICIEL à se comporter conformément aux spécifications, la probabilité d'une telle défaillance doit être supposée de 100 %.

Si le RISQUE de mort ou de BLESSURE GRAVE provenant d'une défaillance logicielle est ensuite ramené à un niveau acceptable (tel que défini par l'ISO 14971) par des mesures de MAITRISE DES RISQUES matérielles, soit en réduisant les conséquences de la défaillance, soit en réduisant la probabilité de mort ou de BLESSURE GRAVE provenant de cette défaillance, la classe de sécurité du logiciel peut être ramenée de C à B. Si le RISQUE de BLESSURE non GRAVE provenant d'une défaillance logicielle est de la même manière ramené à un niveau acceptable par des mesures de MAITRISE DES RISQUES matérielles, la classe de sécurité du logiciel peut être ramenée de B à A.

b) Le FABRICANT doit attribuer à chaque SYSTEME LOGICIEL qui contribue à la mise en œuvre des mesures de MAITRISE DES RISQUES une classe de sécurité du logiciel, fondée sur les effets possibles du PHENOMENE DANGEREUX qui sont couverts par la mesure de MAITRISE DU RISQUE.

c) Le FABRICANT doit consigner la classe de sécurité du logiciel attribuée à chaque SYSTEME LOGICIEL dans le dossier de GESTION DES RISQUES.

d) Lorsqu'un SYSTEME LOGICIEL est décomposé en ELEMENTS LOGICIELS et qu'un ELEMENT LOGICIEL est décomposé en d'autres ELEMENTS LOGICIELS, ces ELEMENTS LOGICIELs doivent hériter de la classe de sécurité du logiciel de l'ELEMENT LOGICIEL initial (ou du SYSTEME LOGICIEL) à moins que le FABRICANT ne justifie par écrit une classification dans une classe différente de sécurité du logiciel. Une telle justification doit expliquer la manière dont les nouveaux ELEMENTS LOGICIELS sont différenciés pour pouvoir être classés séparément.

e) Le FABRICANT doit consigner la classe de sécurité du logiciel de chaque élément LOGICIEL si cette classe est différente de la classe de l'ELEMENT LOGICIEL à partir duquel il a été créé par décomposition.

f) Pour la conformité à la présente norme, lorsqu'un PROCESSUS est exigé pour les ELEMENTS LOGICIELS d'une classe spécifique et que ce PROCESSUS est nécessairement appliqué à un groupe d'ELEMENTS LOGICIELS, le FABRICANT doit utiliser les PROCESSUS et TACHES qui sont exigés par la classe de sécurité de l'ELEMENT LOGICIEL la plus élevée définie dans le groupe à moins que le FABRICANT ne justifie par écrit dans le dossier de GESTION DES RISQUES, l'utilisation d'une classification plus basse.

g) Pour chaque SYSTEME LOGICIEL, les exigences de la classe C doivent être appliquées jusqu'à attribution d'une classe de SECURITE du logiciel.

NOTE Dans les exigences qui suivent, les classes de sécurité du logiciel pour lesquelles l'exigence doit être réalisée sont identifiées en suivant l'exigence sous la forme [Classe...].

5 PROCESSUS de développement du logiciel

5.1 * Planification du développement du logiciel

5.1.1 Plan de développement du logiciel

Le FABRICANT doit établir un(des) plan(s) de développement du logiciel pour entreprendre des ACTIVITES DU PROCESSUS DE DEVELOPPEMENT DU LOGICIEL convenant au domaine d'application, à l'importance et aux classes de sécurité du logiciel du SYSTEME LOGICIEL à développer. Le MODELE DE CYCLE DE VIE DE DEVELOPPEMENT DU LOGICIEL doit être soit complètement défini, soit référencé dans le ou les plans. Le plan doit traiter des éléments suivants:

a) les PROCESSUS à utiliser au cours du développement du SYSTEME LOGICIEL (voir Note 4);

b) les LIVRABLES (y compris la documentation) des ACTIVITES et TACHES;

c) la TRAÇABILITE entre les exigences du SYSTEME, les exigences du logiciel, les essais du SYSTEME LOGICIEL et les mesures de MAITRISE DU RISQUE mis en œuvre dans le logiciel;

d) la gestion de la configuration et des modifications du logiciel, y compris les ELEMENTS DE CONFIGURATION de LOGICIEL DE PROVENANCE INCONNUE (SOUP) utilisés à l'appui du développement; et

e) la résolution des problèmes de logiciel pour le traitement des problèmes détectés dans les PRODUITS LOGICIELS, dans les LIVRABLES et dans les ACTIVITES à chaque étape du cycle de vie.

[Classes A, B, C]

NOTE 1 LE MODELE DU CYCLE DE VIE DE DEVELOPPEMENT DU LOGICIEL peut identifier différents éléments (PROCESSUS, ACTIVITES, TACHES et LIVRABLES) pour différents ELEMENTS LOGICIELS en fonction de la classe de sécurité du logiciel de chaque L'ELEMENT LOGICIEL du SYSTEME LOGICIEL.

NOTE 2 Il est possible que ces ACTIVITES et TACHES se chevauchent ou interagissent et elles peuvent être réalisées de manière itérative ou récursive. La présente norme n'a pas pour but de recommander l'utilisation d'un modèle de cycle de vie spécifique.

NOTE 3 D'autres PROCESSUS sont décrits dans la présente norme indépendamment du PROCESSUS de développement. Ceci ne signifie pas qu'ils doivent être mis en œuvre comme des ACTIVITES et TACHES séparées. Les ACTIVITES et TACHES des autres PROCESSUS peuvent être intégrées dans le PROCESSUS de développement.

NOTE 4 Le plan de développement du logiciel peut référencer des PROCESSUS existants ou en définir de nouveaux.

NOTE 5 Le plan de développement du logiciel peut être intégré dans un plan de développement global du SYSTEME.

5.1.2 Mise à jour du plan de développement logiciel

Le FABRICANT doit mettre à jour le plan au fur et à mesure du développement, le cas échéant. [Classes A, B, C]

5.1.3 Référence du plan de développement du logiciel à la conception et au développement du SYSTEME

a) Le FABRICANT doit référencer les exigences du SYSTEME en tant qu'éléments d'entrée dans le plan de développement du logiciel.

b) Le FABRICANT doit intégrer ou référencer dans le plan de développement du logiciel les procédures de coordination du développement du logiciel ainsi que de validation de la conception et du développement qui sont nécessaires POUR satisfaire aux exigences du 4.1.

[Classes A, B, C]

NOTE Il pourrait ne pas y avoir de différence entre les exigences du SYSTEME LOGICIEL et les exigences du SYSTEME si le SYSTEME LOGICIEL est un SYSTEME autonome (dispositif uniquement logiciel).

5.1.4 Planification des normes, méthodes et outils de développement du logiciel

Le FABRICANT doit inclure ou référencer dans le plan de développement du logiciel:

a) les normes,

b) les méthodes et

c) les outils

associés au développement des ELEMENTS LOGICIELS de classe C. [Classe C]

5.1.5 Planification de l'intégration du logiciel et des essais d'intégration

Le FABRICANT doit inclure ou référencer dans le plan de développement du logiciel un plan d'intégration des ELEMENTS LOGICIELS (y compris les logiciels SOUP) et de réalisation d'essais pendant l'intégration. [Classes B, C]

NOTE Il est admis de combiner en un seul plan et en un seul ensemble d'ACTIVITES, les essais d'intégration et les essais du SYSTEME LOGICIEL.

5.1.6 Planification de la VERIFICATION du logiciel

Le FABRICANT doit inclure ou référencer dans le plan de développement du logiciel les informations suivantes relatives à la VERIFICATION:

a) LES LIVRABLES qui nécessitent une VERIFICATION;

b) les TACHEs de VERIFICATION requises pour chaque ACTIVITE du cycle de vie;

c) les étapes auxquelles les LIVRABLES sont VERIFIES; et

d) les critères d'acceptation de la VERIFICATION des LIVRABLES.

[Classes A, B, C]

5.1.7 Planification de la GESTION DES RISQUES du logiciel

Le FABRICANT doit inclure ou référencer dans le plan de développement du logiciel, un plan relatif à la réalisation des ACTIVITES et TACHES du PROCESSUS DE GESTION DES RISQUES du logiciel, y compris la gestion des RISQUEs liés aux logicielS SOUP. [Classes A, B, C]

NOTE Voir Article 7.

5.1.8 Planification de la documentation

Le FABRICANT doit inclure ou référencer dans le plan de développement du logiciel, les informations relatives aux documents à produire pendant le cycle de vie de développement du logiciel. Pour chaque document identifié ou type de document, les informations suivantes doivent être incluses ou référencées:

a) le titre, le nom ou la convention de désignation;

b) l'objet;

c) l'audience/la diffusion à laquelle le document est destiné; et

d) les procédures et responsabilités de développement, de revue, d'approbation et de modification.

[Classes A, B, C]

5.1.9 Planification de la gestion de configuration du logiciel

Le FABRICANT doit inclure ou référencer les informations relatives à la gestion de la configuration du logiciel dans le plan de développement du logiciel. Les informations de gestion de la configuration du logiciel doivent comprendre ou référencer:

a) les classes, types, catégories ou listes d'éléments à contrôler;

b) les ACTIVITEs et TACHES de gestion de la configuration du logiciel;

c) la ou les organisation(s) chargée(s) de réaliser la gestion de la configuration du logiciel et les ACTIVITEs correspondantes;

d) leur lien avec d'autres organisations telles que celles chargées du développement ou de la maintenance du logiciel;

e) le moment où les éléments doivent être mis sous contrôle de la configuration; et

f) le moment où le PROCESSUS de résolution du problème doit être utilisé.

[Classes A, B, C]

5.1.10 Eléments annexes à contrôler

Les éléments à contrôler doivent inclure les outils, les éléments ou les réglages utilisés pour développer le LOGICIEL DE DISPOSITIF MEDICAL, qui pourraient avoir un impact sur le LOGICIEL DE DISPOSITIF MEDICAL. [Classes B, C]

NOTE De tels éléments comprennent par exemple les versions de programmes de compilation/ de langages d'assemblage, les fichiers makefile, les fichiers séquentiels et les réglages environnementaux spécifiques.

5.1.11 Contrôle de l'ELEMENT DE CONFIGURATION du logiciel avant VERIFICATION

Le FABRICANT doit planifier la mise des ELEMENTS DE CONFIGURATION sous contrôle documenté de la gestion de la configuration avant qu'ils ne soient VERIFIES. [Classes B, C]

5.2 * Analyses des exigences du logiciel

5.2.1 Définition et documentation des exigences du logiciel d'après les exigences du SYSTEME

Pour chaque SYSTEME LOGICIEL du DISPOSITIF MEDICAL, le FABRICANT doit définir et consigner les exigences du SYSTEME LOGICIEL à partir des exigences au niveau du SYSTEME. [Classes A, B, C]

NOTE Il pourrait ne pas y avoir de différence entre les exigences du SYSTEME LOGICIEL et les exigences du SYSTEME si le SYSTEME LOGICIEL est un SYSTEME autonome (dispositif uniquement logiciel).

5.2.2 Teneur des exigences du logiciel

Selon la pertinence pour le LOGICIEL DU DISPOSITIF MEDICAL, le FABRICANT doit inclure dans les exigences du logiciel:

a) les exigences en termes de fonctionnalité et de capacité;

NOTE 1 Les exemples comprennent:

– la performance (par exemple, objet du logiciel, exigences de synchronisation),

– les caractéristiques physiques (par exemple, langage de codage, plate-forme, système d'exploitation),

– l'environnement informatique (par exemple matériel, taille de la mémoire, unité centrale, zone de temps, synchronisation, infrastructure du réseau) dans lequel le logiciel doit fonctionner, et

– le besoin de compatibilité avec des mises à niveau ou des versions multiples de logiciel de provenance inconnue (SOUP) ou d'autres dispositifs.

b) Les éléments d'entrée et de sortie DU SYSTEME LOGICIEL;

NOTE 2 Les exemples comprennent:

- les caractéristiques des données (par exemple, numérique, alphanumérique, format),
- les plages,
- les limites, et
- les valeurs par défaut;

c) Les interfaces entre le SYSTEME LOGICIEL et d'autres SYSTEMES;

d) Les alarmes, avertissements et messages opérateurs générés par le logiciel;

e) Les exigences en matière de SURETE;

NOTE 3 Les exemples comprennent:

- les exigences liées au compromis en matière d'informations sensibles,
- les exigences d'authentification,
- les exigences d'autorisation,
- les exigences d'enregistrement d'audit, et
- les exigences d'intégrité des communications

f) Les exigences de l'ingénierie de l'aptitude à l'utilisation qui sont sensibles aux erreurs humaines et à la formation:

NOTE 4 Les exemples comprennent les exigences liées à:

- l'assistance pour les opérations manuelles,
- les interactions homme-machine,
- les contraintes pour le personnel, et
- les domaines nécessitant une concentration de la part de l'opérateur;

NOTE 5 Les informations relatives aux exigences de l'ingénierie de l'aptitude à l'utilisation sont données dans la CEI 60601-1-6.

g) Les exigences en matière de base de données et de définitions des données;

NOTE 6 Les exemples comprennent:

- le format,
- l'adéquation,
- la fonction.

h) Les exigences d'installation et d'acceptation du logiciel de DISPOSITIF MEDICAL livré au(x) site(s) d'exploitation et de maintenance;

i) Les exigences liées aux méthodes d'exploitation et de maintenance;

j) La documentation utilisateur à élaborer;

k) Les exigences de maintenance par l'utilisateur; et

l) Les exigences réglementaires.

[Classes A, B, C]

NOTE 7 Il est admis que toutes ces exigences ne soient pas disponibles au début du PROCESSUS de développement du logiciel.

NOTE 8 L'ISO/CEI 9126-1 [8] fournit des informations sur les caractéristiques de la qualité qui peuvent être utiles pour la définition des exigences logicielles.

5.2.3 Intégration des mesures de MAITRISE DU RISQUE dans les exigences du logiciel

Le FABRICANT doit inclure dans les exigences les mesures de MAITRISE DU RISQUE mises en œuvre dans le logiciel pour tenir compte des défaillances matérielles et des éventuels défauts du logiciel, en fonction du DISPOSITIF MEDICAL. [Classes B, C]

NOTE Il est admis que ces exigences ne soient pas disponibles au début du PROCESSUS de développement du logiciel et peuvent changer au fur et à mesure de la conception du logiciel et de la définition des mesures de MAITRISE DU RISQUE.

5.2.4 Ré-EVALUATION de l'ANALYSE DU RISQUE du DISPOSITIF MEDICAL

Une fois les exigences du logiciel établies, le FABRICANT doit ré-EVALUER et si nécessaire mettre à jour l'ANALYSE DE RISQUE du DISPOSITIF MEDICAL. [Classes A, B, C]

5.2.5 Mise à jour des exigences du SYSTEME

Le FABRICANT doit s'assurer que les exigences existantes, y compris les exigences du SYSTEME sont ré-EVALUEES et correctement mises à jour en fonction des résultats de l'ACTIVITE d'analyse des exigences du logiciel. [Classes A, B, C]

5.2.6 Vérification des exigences du logiciel

Le FABRICANT doit vérifier et consigner que les exigences du logiciel:

a) mettent en œuvre les exigences du SYSTEME, y compris celles liées à la MAITRISE DU RISQUE;

b) ne se contredisent pas;

c) sont exprimées en termes univoques:

d) sont indiquées en des termes qui permettent d'établir les critères et les performances des essais de manière à s'assurer que ces critères sont remplis;

e) peuvent être identifiées de manière unique; et

f) leur TRAÇABILITE aux exigences du SYSTEME ou autre source est assurée;

[Classes A, B, C]

NOTE La présente norme n'exige pas le recours à un langage de spécification formel.

5.3 * Conception ARCHITECTURALE du logiciel

5.3.1 Conversion des exigences du logiciel en ARCHITECTURE

Le FABRICANT doit transformer les exigences du LOGICIEL DE DISPOSITIF MEDICAL en une ARCHITECTURE documentée décrivant la structure du logiciel et identifiant les éléments LOGICIELS. [Classes B, C]

5.3.2 Elaboration d'une ARCHITECTURE pour les interfaces d'ELEMENTS LOGICIELS

Le FABRICANT doit élaborer et documenter une ARCHITECTURE pour les interfaces entre les ELEMENTS LOGICIELS et les composants externes aux ELEMENTS LOGICIELS (tant logiciels que matériels), ainsi qu'entre les ELEMENTS LOGICIELS proprement dits. [Classes B, C]

5.3.3 Spécification des exigences fonctionnelles et de performance des ELEMENTS LOGICIELS SOUP

Si un ELEMENT LOGICIEL est identifié comme étant SOUP, le FABRICANT doit spécifier les exigences fonctionnelles et de performance dudit élément SOUP qui sont nécessaires à son usage prévu. [Classes B, C]

5.3.4 Spécification des matériels et des logiciels SYSTEME nécessaires à l'ELEMENT LOGICIEL SOUP

Si uN ELEMENT LOGICIEL est identifié comme étant SOUP, le FABRICANT doit spécifier les matériels et logiciels SYSTEME nécessaires pour assurer le fonctionnement correct du logiciel SOUP. [Classes B, C]

NOTE Ces informations comprennent, par exemple, le type et la vitesse du processeur, le type et la taille de la mémoire, le type de SYSTEME logiciel, les exigences relatives au logiciel de communication et d'affichage.

5.3.5 Identification des séparations nécessaires à la MAITRISE DU RISQUE

Le FABRICANT doit identifier les séparations entre ELEMENTS LOGICIELS qui sont essentiels pour la MAITRISE DU RISQUE et indiquer la méthode permettant de s'assurer que la séparation est efficace. [Classe C]

NOTE Un exemple de séparation est l'exécution des ELEMENTS LOGICIELS sur différents processeurs. L'efficacité de la séparation peut être assurée en évitant tout partage de ressources entre les processeurs.

5.3.6 Vérification de l'ARCHITECTURE du logiciel

Le FABRICANT doit vérifier et consigner que:

a) l'ARCHITECTURE du logiciel permet une mise en œuvre deS EXIGENCES DU SYSTEME et des logiciels, y compris celles liées à la MAITRISE DU RISQUE;

b) l'ARCHITECTURE du logiciel est capable de prendre en charge les interfaces entre ELEMENTS LOGICIELS ainsi qu'entre ELEMENTS LOGICIELS et matériels; et

c) l'ARCHITECTURE du DISPOSITIF MEDICAL assure le fonctionnement correct des éventuels ELEMENTS LOGICIELS SOUP utilisés.

[Classes B, C]

5.4 * Conception détaillée du logiciel

5.4.1 Décomposition de l'ARCHITECTURE des LOGICIELS en UNITES LOGICIELLES

Le FABRICANT doit affiner l'ARCHITECTURE logicielle jusqu'à ce qu'elle soit représentée par les UNITES LOGICIELLES. [Classes B, C]

5.4.2 Elaboration de la conception détaillée de chaque UNITE LOGICIELLE

Le FABRICANT doit élaborer et documenter une conception détaillée de chaque UNITE LOGICIELLE de l'ELEMENT LOGICIEL. [Classe C]

5.4.3 Elaboration de la conception détaillée pour les interfaces

Le FABRICANT doit élaborer et documenter une conception détaillée des interfaces éventuelles entre l'UNITE LOGICIELLE et les composants externes (matériels et logiciels), ainsi que pour les interfaces entre UNITES LOGICIELLES. [Classe C]

5.4.4 Vérification de la conception détaillée

Le FABRICANT doit vérifier et consigner que la conception détaillée du logiciel:

a) met en œuvre l'ARCHITECTURE du logiciel; et

b) est exempte de contradiction avec l'ARCHITECTURE du logiciel.

[Classe C]

5.5 * Mise en œuvre et vérification des UNITES LOGICIELLES

5.5.1 Mise en œuvre de chaque UNITE LOGICIELLE

Le FABRICANT doit mettre en œuvre chaque UNITE LOGICIELLE. [Classes A, B, C]

5.5.2 Etablissement du PROCESSUS DE VERIFICATION DES UNITES LOGICIELLES

Le FABRICANT doit établir des stratégies, méthodes et procédures pour la vérification de chaque UNITE LOGICIELLE. Lorsque la VERIFICATION est effectuée sur la base d'essais, les procédures d'essai doivent être EVALUEES pour correction. [Classes B, C]

NOTE Il est admis de combiner en un seul plan et ensemble d'ACTIVITES, LES ESSAIS D'INTEGRATION ET LES ESSAIS DU SYSTEME LOGICIEL.

5.5.3 Critères d'acceptation de l'UNITE LOGICIELLE

Le FABRICANT doit établir des critères d'acceptation pour les UNITES LOGICIELLES avant intégration dans des ELEMENTS LOGICIELS plus grands et s'assurer que les UNITES LOGICIELLES remplissent ces critères d'acceptation.

NOTE Exemples de critères d'acceptation:

– le code du logiciel met-il correctement en œuvre les exigences, y compris les mesures de MAITRISE DU RISQUE ?

– le code du logiciel est-il en contradiction avec les interfaces décrites dans les documents de conception détaillée de l'unité LOGICIELLE ?

– le code du logiciel est-il conforme aux procédures de programmation ou normes de codage établies ?

5.5.4 Critères supplémentaires d'acceptation de l'UNITE LOGICIELLE

Lorsqu'il participe à la conception, LE FABRICANT doit, selon le cas, introduire des critères supplémentaires d'acceptation suivants:

a) du séquencement approprié des événements;

b) du flux des données et du contrôle;

c) de l'affectation des ressources planifiées;

d) de la définition et détection des erreurs et reprise après erreur;

e) de l'initialisation des variables;

f) des autodiagnostics;

g) de la gestion de la mémoire et des dépassements de capacité de la mémoire; et

h) des conditions limites.

[Classe C]

5.5.5 VERIFICATION de l'UNITE LOGICIELLE

Le FABRICANT doit réaliser la VERIFICATION de l'UNITE LOGICIELLE et documenter les résultats.

[Classes B, C]

5.6 * Intégration et essai d'intégration du logiciel

5.6.1 Intégration des UNITES LOGICIELLES

Le FABRICANT doit intégrer les UNITES LOGICIELLES conformément au plan d'intégration (voir 5.1.5). [Classes B, C]

5.6.2 Vérification de l'intégration du logiciel

Le FABRICANT doit vérifier et enregistrer les aspects suivants de l'intégration du logiciel conformément au plan d'intégration (voir 5.1.5):

a) Les UNITES LOGICIELLES ont été intégrées dans les ELEMENTS LOGICIELS dans le SYSTEME LOGICIEL, et

b) Les éléments matériels, ELEMENTS LOGICIELS et l'aide aux opérations manuelles (par exemple: interface homme machine, les menus d'aide en ligne, la reconnaissance vocale, les commandes vocales) du SYSTEME correspondant ont bien été intégrés au SYSTEME.

[Classes B, C]

NOTE Cette VERIFICATION concerne uniquement l'intégration des éléments, conformément au plan, et non leur performance prévue. Cette VERIFICATION est en général réalisée sous forme d'inspection.

5.6.3 Essai du logiciel intégré

Le FABRICANT doit soumettre à des essais les ELEMENTS LOGICIELS intégrés conformément au plan d'intégration (voir 5.1.5) et documenter les résultats correspondants. [Classes B, C]

5.6.4 Teneur des essais d'intégration

Pour les essais d'intégration du logiciel, le FABRICANT doit s'assurer que l'ELEMENT LOGICIEL intégré s'exécute comme prévu.

[Classes B, C]

NOTE 1 Sont à considérer, par exemple:

- la fonctionnalité requise du logiciel;
- la mise en œuvre des mesures de MAITRISE DU RISQUE;
- la synchronisation et autre comportement spécifié;
- le fonctionnement spécifié des interfaces internes et externes; et
- les essais dans des conditions anormales incluant le mauvais usage prévisible.

NOTE 2 Il est admis de combiner en un seul plan et ensemble d'ACTIVITEs, les essais d'intégration et les essais du SYSTEME LOGICIEL.

5.6.5 Vérification des procédures d'essais d'intégration

Le FABRICANT doit EVALUER les procédures d'essais d'intégration pour s'assurer de leur bien fondé. [Classes B, C]

5.6.6 Réalisation d'ESSAIS DE REGRESSION

Lorsque des ELEMENTS LOGICIELS sont intégrés, le FABRICANT doit réaliser un ESSAI DE REGRESSION approprié pour démontrer que des défauts n'ont pas été introduits dans le logiciel précédemment intégré. [Classes B, C]

5.6.7 Contenu de l'enregistrement des essais d'intégration

Le FABRICANT doit:

a) documenter les résultats d'essai (réussite/échec ainsi que la liste des ANOMALIES);

b) conserver suffisamment d'enregistrements pour permettre la reproduction de l'essai; et

c) identifier le contrôleur chargé d'effectuer l'essai.

[Classes B, C]

NOTE L'exigence b) pourrait être mise en œuvre en conservant par exemple:

- les spécifications d'essais montrant les actions requises et les résultats attendus,
- des enregistrements de l'équipement,
- des enregistrements de l'environnement d'essai (y compris les outils logiciels).

5.6.8 Utilisation du PROCESSUS de résolution des problèmes de logiciel

Le FABRICANT doit intégrer les ANOMALIES décelées lors de l'intégration et des essais d'intégration du logiciel dans un PROCESSUS de résolution des problèmes de logiciel [Classes B, C]

NOTE Voir Article 9.

5.7 * Essais du SYSTEME LOGICIEL

5.7.1 Etablissement d'essais pour les exigences du logiciel

Le FABRICANT doit établir et réaliser un ensemble d'essais exprimé en stimuli d'entrée, résultats attendus, critères de réussite/échec et en procédures d'exécution des essais du SYSTEME LOGICIEL, de telle sorte que toutes les exigences du logiciel soient couvertes. [Classes B, C]

NOTE 1 Il est admis de combiner en un seul plan et ensemble d'ACTIVITES, les ESSAIS D'INTEGRATION et les essais du SYSTEME LOGICIEL. Il est également admis de soumettre les exigences logicielles à des essais au cours de phases antérieures.

NOTE 2 Il peut être réalisé, non seulement des essais séparés pour chaque exigence, mais aussi des essais de combinaison d'exigences, en particulier s'il existe des dépendances entre les exigences.

5.7.2 Utilisation du PROCESSUS de résolution des problèmes de logiciel

Le FABRICANT doit intégrer dans un PROCESSUS de résolution des problèmes de logiciel les ANOMALIES décelées au cours de l'essai du SYSTEME LOGICIEL. [Classes B, C]

5.7.3 Contre-essais après modifications

Lorsque des modifications sont effectuées pendant les essais du SYSTEME LOGICIEL, le FABRICANT doit:

a) recommencer les essais, effectuer des essais modifiés ou des essais supplémentaires, selon le cas, afin de vérifier l'efficacité de la modification pour la correction du problème;

b) effectuer des essais appropriés afin de démontrer que des effets secondaires non prévus n'ont pas été introduits; et

c) réaliser les ACTIVITES pertinentes de GESTION DES RISQUES comme définies en 7.4.

[Classes B, C]

5.7.4 Vérification des essais du SYSTEME LOGICIEL

Le FABRICANT doit vérifier que:

a) les stratégies de VERIFICATION utilisées sont appropriées et les procédures d'essai sont appropriées;

b) les procédures d'essai du SYSTEME LOGICIEL sont tracées vis-à-vis des exigences du logiciel;

c) toutes les exigences du logiciel ont été soumises à des essais ou VERIFIEES par ailleurs; et

d) les résultats d'essai satisfont aux critères de réussite/échec.

[Classes B, C]

5.7.5 Teneur des enregistrements d'essai du SYSTEME LOGICIEL

Le FABRICANT doit:

a) documenter les résultats d'essai (réussite/échec ainsi que la liste des ANOMALIES);

b) conserver suffisamment d'enregistrements pour permettre la reproduction de l'essai; et

c) identifier le contrôleur chargé d'effectuer l'essai.

[Classes B, C]

NOTE L'exigence b) pourrait être mise en œuvre en conservant par exemple:

– les spécifications du jeu d'essais montrant les actions requises et les résultats attendus;

– des enregistrements du matériel d'essai; et

– des enregistrements de l'environnement d'essai (y compris les outils logiciels).

5.8 * Diffusion du logiciel

5.8.1 Assurance de l'achèvement de la VERIFICATION du logiciel

Le FABRICANT doit s'assurer, avant diffusion du logiciel, que la VERIFICATION du logiciel est terminée et que les résultats ont été évalués. [Classes B, C]

5.8.2 Consignation des ANOMALIES résiduelles connues

Le FABRICANT doit documenter toutes les ANOMALIES résiduelles connues. [Classes B, C]

5.8.3 Évaluation des ANOMALIES résiduelles connues

Le FABRICANT doit s'assurer que toutes les ANOMALIES résiduelles connues ont été évaluées pour s'assurer qu'elles ne contribuent pas à un RISQUE inacceptable. [Classes B, C]

5.8.4 Consignation des VERSIONS diffusées

Le FABRICANT doit documenter la VERSION du PRODUIT LOGICIEL qui est diffusée. [Classes A, B, C]

5.8.5 Consignation de la manière dont le logiciel diffusé a été créé

Le FABRICANT doit documenter la procédure et l'environnement utilisés pour créer le logiciel diffusé. [Classes B, C]

5.8.6 Assurance de l’achèvement complet des ACTIVITES et des TACHES

Le FABRICANT doit s'assurer que toutes les ACTIVITES et TACHES sont complètement achevées et que la documentation associée est complète. [Classes B, C]

5.8.7 Archivage du logiciel

Le FABRICANT doit archiver:

a) le PRODUIT LOGICIEL et les ELEMENTS DE CONFIGURATION; et

b) la documentation

pendant au moins une période déterminée comme étant la plus longue entre la durée de vie du dispositif telle que définie par le FABRICANT et un laps de temps spécifié par les exigences réglementaires pertinentes. [Classes B, C]

5.8.8 Assurance de la reproductibilité du logiciel diffusé

Le FABRICANT doit établir des procédures pour s’assurer que le PRODUIT LOGICIEL diffusé est bien livré de manière fiable au point d'utilisation, sans corruption ni changement non-autorisé. Ces procédures doivent concerner la production et le maniement des supports contenant le PRODUIT LOGICIEL, incluant selon le cas:

- la réplication,
- l’étiquetage,
- l’emballage,
- la protection,
- le stockage, et
- la livraison.

[Classes B, C]

6 PROCESSUS de maintenance du logiciel

6.1 * Etablissement du plan de maintenance du logiciel

Le FABRICANT doit établir un(des) plan(s) de maintenance du logiciel afin d'entreprendre les ACTIVITES et TACHES du PROCESSUS de maintenance. Le plan doit traiter des éléments suivants:

a) des procédures de:
   - réception,
   - documentation,
   - évaluation,
   - résolution et
   - suivi

   des retours d'information survenant après la diffusion du logiciel de DISPOSITIF MEDICAL;

b) les critères permettant de déterminer si ces retours d'information constituent effectivement des problèmes ;

c) l'utilisation du PROCESSUS de GESTION DES RISQUES du logiciel;

d) l'utilisation du PROCESSUS de résolution des problèmes du logiciel afin d'analyser et résoudre les problèmes après diffusion du LOGICIEL DE DISPOSITIF MEDICAL;

e) l'utilisation du PROCESSUS de gestion de la configuration du logiciel (Article 8) pour la gestion des modifications au SYSTEME existant; et

f) les procédures d'évaluation et de mise en œuvre:
   - des mises à niveau,
   - des corrections de bogue,
   - des patches (rustines) et
   - de l'obsolescence

   des logiciels SOUP.

[Classes A, B, C]

6.2 * Analyse des problèmes et des modifications

6.2.1 Consignation et EVALUATION des retours d'information

6.2.1.1 Contrôle des retours d'information

Le FABRICANT doit contrôler tout retour d'information sur le PRODUIT LOGICIEL diffusé tant à l'intérieur de sa propre organisation que de la part des utilisateurs. [Classes A, B, C]

6.2.1.2 Consignation et EVALUATION des retours d'information

Les retours d'information doivent être documentés et EVALUES pour déterminer s'il y a un problème dans un PRODUIT LOGICIEL diffusé. Tout problème de ce type doit être enregistré comme RAPPORT DE PROBLEME (voir Article 9) Les RAPPORTS DE PROBLEME doivent comprendre les événements préjudiciables réels ou potentiels, ainsi que les écarts par rapport aux spécifications. [Classes A, B, C]

6.2.1.3 Évaluation des influences des RAPPORTS DE PROBLEME sur la SECURITE

Chaque RAPPORT DE PROBLEME doit être évalué afin de déterminer la manière dont il affecte la SECURITE d'un PRODUIT LOGICIEL diffusé et si une modification du PRODUIT LOGICIEL diffusé est nécessaire pour traiter le problème. [Classes A, B, C]

6.2.2 Utilisation du PROCESSUS de résolution des problèmes du logiciel

Le FABRICANT doit utiliser le PROCESSUS de résolution des problèmes du logiciel (voir l'Article 9) pour traiter les RAPPORTS DE PROBLEME. [Classes A, B, C]

NOTE Une fois cette ACTIVITE terminée, il convient de connaître toute modification de classe de sécurité dans le SYSTEME LOGICIEL ou ses ELEMENTS LOGICIELS.

6.2.3 Analyse des DEMANDES DE MODIFICATION

En plus de l'analyse exigée par l'Article 9, le FABRICANT doit analyser chaque DEMANDE DE MODIFICATION afin d'évaluer ses effets sur l'organisation, sur les PRODUITS LOGICIELS diffusés, ainsi que les SYSTEMES auxquels il est interfacé. [Classe B, C]

6.2.4 Approbation des DEMANDES DE MODIFICATION

Le FABRICANT doit EVALUER et approuver les DEMANDES DE MODIFICATION qui modifient les PRODUITS LOGICIELS diffusés. [Classes A, B, C]

6.2.5 Communication aux utilisateurs et aux organismes de réglementation

Le FABRICANT doit identifier les DEMANDES DE MODIFICATION approuvées qui affectent les PRODUITS LOGICIELS diffusés.

Si la réglementation locale l'exige, le FABRICANT doit informer les utilisateurs et les organismes de réglementation:

a) de tout problème affectant les PRODUITS LOGICIELS diffusés et les conséquences de la poursuite de leur utilisation sans modification; et

b) de la nature de toutes les modifications disponibles pour les produits LOGICIELS diffusés ainsi que la manière d'obtenir et d'installer ces modifications.

[Classes A, B, C]

6.3 * Mise en œuvre de la modification

6.3.1 Utilisation d'un PROCESSUS établi pour mettre en œuvre la modification

Le FABRICANT doit utiliser le PROCESSUS de développement du logiciel (voir l'Article 5) ou un PROCESSUS de maintenance établi pour mettre en œuvre les modifications. [Classes A, B, C]

NOTE Pour les exigences concernant la GESTION DES RISQUES des modifications du logiciel, voir 7.4.

6.3.2 Rediffusion du SYSTEME LOGICIEL modifié

Le FABRICANT doit diffuser les SYSTEMES LOGICIELS modifiés conformément à 5.8. Les modifications peuvent être diffusées dans le cadre d'une rediffusion complète d'un SYSTEME LOGICIEL ou comme un kit de modifications comprenant les ELEMENTS LOGICIELS modifiés et les outils nécessaires pour installer les modifications pour un SYSTEME LOGICIEL existant. [Classes A, B, C]

7 * PROCESSUS DE GESTION DES RISQUES du logiciel

7.1 * Analyse du logiciel en termes de contribution à des situations dangereuses

7.1.1 Identification des ELEMENTS LOGICIELS qui pourraient contribuer à une situation dangereuse

Le FABRICANT doit identifier les ELEMENTS LOGICIELS qui pourraient contribuer à une situation dangereuse identifiée par l'ACTIVITE d'ANALYSE DU RISQUE du DISPOSITIF MEDICAL comme défini dans l'ISO 14971 (voir 4.2). [Classes B, C]

NOTE La situation dangereuse pourrait être le résultat direct d'une défaillance de logiciel ou le résultat de la défaillance d'une mesure de MAITRISE DU RISQUE mise en œuvre dans le logiciel.

7.1.2 Identification des causes potentielles de contribution à une situation dangereuse

Le FABRICANT doit identifier les causes potentielles de la contribution de l'ELEMENT LOGICIEL identifié ci-dessus à une situation dangereuse.

Le FABRICANT doit tenir compte des causes potentielles, incluant le cas échéant:

a) la spécification incorrecte ou incomplète de la fonctionnalité;

b) les défauts logiciels dans la fonctionnalité de l'ELEMENT LOGICIEL identifié;

c) la défaillance ou résultat inattendu du logiciel SOUP;

d) les défaillances matérielles ou autres défauts logiciels qui pourraient donner lieu à un fonctionnement imprévisible du logiciel; et

e) un mauvais usage raisonnablement prévisible.

[Classes B, C]

7.1.3 ÉVALUATION des listes publiées D'ANOMALIES SOUP

Si une défaillance ou des résultats inattendus d'un logiciel SOUP est une cause potentielle de la contribution d'un ELEMENT LOGICIEL à une situation dangereuse, le FABRICANT doit au minimum EVALUER toute liste d'ANOMALIES publiée par le fournisseur de l'élément du logiciel SOUP concernant la version de l'élément du logiciel SOUP utilisée dans le DISPOSITIF MEDICAL, afin de déterminer si l'une des ANOMALIES connues entraîne une séquence d'événements qui pourrait donner lieu à une situation dangereuse. [Classes B, C]

7.1.4 Consignation des causes potentielles

Le FABRICANT doit documenter dans le DOSSIER DE GESTION DES RISQUES les causes potentielles de la contribution de l'élément LOGICIEL à une situation dangereuse (voir l'ISO 14971). [Classes B, C]

7.1.5 Consignation des séquences d'événements

Le FABRICANT doit documenter, dans le dossier de GESTION DES RISQUES, les séquences d'événements qui pourraient entraîner une situation dangereuse, telles qu'identifiées au 7.1.2. [Classes B, C]

7.2 Mesures DE MAITRISE DU RISQUE

7.2.1 Définition des mesures de MAITRISE DU RISQUE

Le FABRICANT doit définir et documenter des mesures de MAITRISE DU RISQUE pour chaque cause potentielle de l'ELEMENT LOGICIEL contribuant à une situation dangereuse documentée dans le DOSSIER DE GESTION DES RISQUES. [Classe B, C]

NOTE Les mesures de MAITRISE DU RISQUE peuvent être mises en œuvre dans le matériel, dans le logiciel, dans l'environnement de travail ou comme une instruction destinée à l'utilisateur.

7.2.2 Mesures de MAITRISE DU RISQUE mises en œuvre dans le logiciel

Si une mesure de MAITRISE DU RISQUE est mise en œuvre comme faisant partie des fonctions d'un ELEMENT LOGICIEL, le FABRICANT doit:

a) inclure la mesure de MAITRISE DU RISQUE dans les exigences de logiciel;

b) attribuer une classe de SECURITE du logiciel à l'élément LOGICIEL sur la base des effets possibles du PHENOMENE DANGEREUX que contrôle la MAITRISE DU RISQUE; et

c) développer l'ELEMENT LOGICIEL conformément à l'Article 5.

[Classes B, C]

NOTE Cette exigence fournit de plus amples détails sur les exigences de la MAITRISE DU RISQUE de l'ISO 14971.

7.3 VERIFICATION des mesures de MAITRISE DU RISQUE

7.3.1 Vérification des mesures de MAITRISE DU RISQUE

La mise en œuvre de chacune des mesures de MAITRISE DU RISQUE indiquées en 7.2 doit être VERIFIEE, et cette VERIFICATION doit être documentée. [Classes B, C]

7.3.2 Consignation de toutes nouvelles séquences d'événements

Si une mesure de MAITRISE DU RISQUE est mise en œuvre comme ELEMENT LOGICIEL, le FABRICANT doit EVALUER la mesure de MAITRISE DU RISQUE afin d'identifier et de documenter, dans le DOSSIER DE GESTION DES RISQUES, les éventuelles nouvelles séquences d'événements qui pourraient entraîner une situation dangereuse. [Classes B, C]

7.3.3 Consignation de la TRAÇABILITE

Le FABRICANT doit documenter la TRAÇABILITE des DANGERS liés au logiciel selon le cas:

a) de la situation dangereuse à l'ELEMENT LOGICIEL;

b) de l'ELEMENT LOGICIEL à la cause logicielle spécifique;

c) de la cause logicielle à la mesure de MAITRISE DU RISQUE; et

d) de la mesure dE MAITRISE DU RISQUE à la VERIFICATION de la mesure de MAITRISE DU RISQUE.

[Classes B, C]

NOTE Voir l'ISO 14971 – Rapport DE GESTION DES RISQUES.

7.4 GESTION DES RISQUES des modifications du logiciel

7.4.1 Analyse des modifications apportées au LOGICIEL DE DISPOSITIF MEDICAL en termes de SECURITE

Le FABRICANT doit analyser les modifications apportées au LOGICIEL DE DISPOSITIF MEDICAL (y compris les logiciels SOUP) afin de déterminer si:

a) des causes potentielles supplémentaires contribuant à une situation dangereuse sont introduites; et

b) des mesures supplémentaires de MAITRISE DU RISQUE du logiciel sont nécessaires.

[Classes A, B, C]

7.4.2 Analyse de l'impact des modifications apportées au logiciel sur les mesures existantes de MAITRISE DU RISQUE

Le FABRICANT doit analyser les modifications apportées au logiciel, y compris celles apportées au logiciel SOUP, afin de déterminer si la modification du logiciel pourrait interférer avec des mesures existantes de MAITRISE DU RISQUE. [Classes B, C]

7.4.3 Réalisation des ACTIVITES DE GESTION DES RISQUES sur la base des analyses

Le FABRICANT doit réaliser les ACTIVITEs pertinentes de GESTION DES RISQUES définies en 7.1, 7.2 et 7.3 sur la base de ces analyses. [Classes B, C]

8 * PROCESSUS de gestion de configuration du logiciel

8.1 * Identification de la configuration

8.1.1 Etablissement des moyens d'identification des ELEMENTS DE CONFIGURATION

Le FABRICANT doit établir un plan pour l'identification univoque des ELEMENTS DE CONFIGURATION et leurs VERSIONS à maîtriser dans le cadre du projet. Ce plan doit inclure les autres PRODUITS LOGICIELS ou entités tels que les logiciels SOUP et la documentation. [Classes A, B, C]

8.1.2 Identification des logiciels SOUP

Pour chaque ELEMENT DE CONFIGURATION de logiciel SOUP utilisé, y compris les bibliothèques standards, le FABRICANT doit documenter:

a) le titre,

b) le FABRICANT, et

c) la désignation unique du logiciel SOUP

de chaque ELEMENT DE CONFIGURATION de logiciel SOUP à utiliser. [Classes A, B, C]

NOTE La désignation unique du logiciel SOUP pourrait être par exemple, une VERSION, une date de diffusion, un numéro de correctif ou une désignation de mise à niveau.

8.1.3 Identification de la documentation de configuration du SYSTEME

Le FABRICANT doit documenter l'ensemble des éléments de CONFIGURATION et leurs VERSIONS qui constituent la configuration du SYSTEME LOGICIEL. [Classes A, B, C]

8.2 * Maîtrise des modifications

8.2.1 Approbation des DEMANDES DE MODIFICATION

Le FABRICANT ne doit modifier des ELEMENTS DE CONFIGURATION qu'en réponse à une DEMANDE DE MODIFICATION approuvée. [Classes A, B, C]

NOTE 1 La décision d'approuver une DEMANDE DE MODIFICATION peut être intégrée au PROCESSUS de maîtrise des modifications ou faire partie d'un autre PROCESSUS. Le présent paragraphe exige uniquement qu'une modification soit approuvée avant sa mise en œuvre.

NOTE 2 Différents PROCESSUS d'acceptation peuvent être utilisés pour les DEMANDES DE MODIFICATION à différentes étapes du cycle de vie, comme indiqué dans les plans, voir 5.1.1 e) et 6.1 e).

8.2.2 Mise en œuvre des modifications

Le FABRICANT doit mettre en œuvre la modification comme spécifié dans la DEMANDE DE MODIFICATION. Le FABRICANT doit identifier et réaliser toute ACTIVITE qu'il est nécessaire de répéter du fait de la modification, y compris les changements de la classification de SECURITE DU LOGICIEL des SYSTEMES LOGICIELS et des ELEMENTS LOGICIELS. [Classe A, B, C]

NOTE Le présent paragraphe précise la manière dont il convient de mettre en œuvre la modification pour assurer une maîtrise appropriée des modifications. Il n'implique en aucune manière que la mise en œuvre fait partie intégrante du PROCESSUS de maîtrise des modifications. Il est recommandé que la mise en œuvre utilise des PROCESSUS planifiés, voir 5.1.1 e) et 6.1 e).

8.2.3 Vérification des modifications

Le FABRICANT doit vérifier la modification, y compris la répétition de toute VERIFICATION invalidée par la modification et la prise en compte du 5.7.3 et du 9.7. [Classes A, B, C]

NOTE Ce paragraphe exige uniquement que les modifications soient vérifiées. Il n'implique en aucune manière que la VERIFICATION fasse partie intégrante du PROCESSUS de maîtrise des modifications. Il est recommandé que la VERIFICATION utilise des PROCESSUS planifiés, voir 5.1.1 e) et 6.1 e).

8.2.4 Prévision des moyens de TRAÇABILITE de la modification

Le FABRICANT doit créer un enregistrement d'audit permettant à chaque:

a) DEMANDE DE MODIFICATION;

b) RAPPORT DE PROBLEME pertinent; et

c) approbation de la DEMANDE DE MODIFICATION

d'être tracée. [Classes A, B, C]

8.3 * Documentation relative à l'état de la configuration

Le FABRICANT doit conserver des enregistrements récupérables de l'historique des ELEMENTS DE CONFIGURATION maîtrisés y compris la configuration du SYSTEME. [Classes A, B, C]

9 * PROCESSUS de résolution de problème logiciel

9.1 Elaboration des RAPPORTS DE PROBLEME

Le FABRICANT doit rédiger un RAPPORT DE PROBLEME pour chaque problème détecté dans un PRODUIT LOGICIEL. Les RAPPORTS DE PROBLEME doivent être classés comme suit:

a) le type;

EXEMPLE 1 correctif, préventif ou adaptation à un nouvel environnement

b) le domaine d'application; et

EXEMPLE 2 étendue de la modification, nombre de modèles de dispositifs concernés, accessoires pris en charge concernés, ressources impliquées, temps nécessaire pour la modification

c) la criticité.

EXEMPLE 3 effet sur les performances, LA SECURITE ou LA SURETE

[Classes A, B, C]

NOTE Les problèmes peuvent être décelés avant ou après diffusion, à l'intérieur ou à l'extérieur de l'organisation du FABRICANT.

9.2 Etude du problème

Le FABRICANT doit:

a) étudier le problème et si possible en identifier les causes;

b) EVALUER la pertinence du problème en termes de SECURITE en utilisant le PROCESSUS de GESTION DES RISQUEs du logiciel (Article 7);

c) documenter le résultat de la recherche et de l'évaluation; et

d) déclencher la ou les DEMANDES DE MODIFICATION nécessaires pour corriger le problème, ou justifier le fait de n'entreprendre aucune action.

[Classes A, B, C]

NOTE Il n'est pas nécessaire qu'un problème ait été corrigé pour que le FABRICANT soit conforme au PROCESSUS de résolution des problèmes du logiciel, si le problème ne concerne pas la SECURITE.

9.3 Information des parties concernées

Le FABRICANT doit informer les parties concernées de l'existence du problème, selon le cas. [Classes A, B, C]

NOTE Les problèmes peuvent être découverts avant ou après la diffusion, à l'intérieur de l'organisation du FABRICANT ou à l'extérieur. Le FABRICANT détermine les parties concernées en fonction de la situation.

9.4 Utilisation du processus de la maîtrise des modifications

Le FABRICANT doit approuver et mettre en œuvre toutes les DEMANDES DE MODIFICATION, en observant les exigences du processus de la maîtrise des modifications (voir 8.2). [Classes A, B, C]

9.5 Conservation des enregistrements

Le FABRICANT doit conserver des enregistrements des RAPPORTS DE PROBLEMES et de leur résolution y compris leur VERIFICATION.

Le FABRICANT doit mettre à jour le dossier de GESTION DES RISQUES selon le cas (voir 7.4). [Classes A, B, C]

9.6 Analyse de tendance pour les problèmes

Le FABRICANT doit effectuer une analyse permettant de détecter les tendances dans les RAPPORTS DE PROBLEMES. [Classes A, B, C]

9.7 VERIFICATION de la résolution des problèmes du logiciel

Le FABRICANT doit vérifier les résolutions de problèmes afin de s'assurer:

a) que le problème a été résolu et que le RAPPORT DE PROBLEMES a été clos;

b) que les tendances préjudiciables ont été inversées;

c) que les DEMANDES DE MODIFICATION ont été mises en œuvre dans les PRODUITS LOGICIELS et ACTIVITES concernées; et

d) que des problèmes supplémentaires n'ont pas été introduits.

[Classes A, B, C]

9.8 Teneur de la documentation d'essai

Lors d'essais, de contre-essais ou d'essais de régression des ELEMENTS LOGICIELS et SYSTEMES, suite à une modification, le FABRICANT doit inclure dans la documentation d'essai:

a) les résultats d'essai;

b) les ANOMALIES décelées;

c) la VERSION du logiciel soumis à l'essai;

d) les configurations d'essai pertinentes pour le matériel et le logiciel;

e) les outils d'essai pertinents;

f) la date de l'essai; et

g) l'identification du contrôleur ayant effectué l'essai.

[Classes A, B, C]

Annexe A
(informative)

Justification des exigences de la présente norme

La présente annexe fournit une justification des articles de la présente norme.

A.1 Justification du raisonnement

La principale exigence de la présente norme est qu'un ensemble de PROCESSUS doit être suivi pour le développement et la maintenance des LOGICIELS DE DISPOSITIFS MEDICAUX et que le choix des PROCESSUS soit adapté aux RISQUES encourus par le patient et autre personne concernée. Ceci procède de la conviction selon laquelle les essais de logiciel ne sont pas suffisants pour déterminer que son fonctionnement est sûr.

Les PROCESSUS exigés par la présente norme s'inscrivent dans deux catégories:

– Les PROCESSUS qui sont exigés pour déterminer les RISQUES résultant du fonctionnement de chaque ELEMENT LOGICIEL dans le logiciel;

– Les PROCESSUS qui sont exigés pour atteindre un taux de probabilité de défaut de logiciel suffisamment bas pour chaque ELEMENT LOGICIEL, choisi sur la base de la détermination desdits RISQUES.

La présente norme exige que la première catégorie soit appliquée à tout LOGICIEL DE DISPOSITIF MÉDICAL et la seconde catégorie porte sur des ELEMENTS LOGICIELS choisis.

Ainsi, il convient que toute revendication de conformité à la présente norme inclue une analyse des RISQUES écrite qui identifie les séquences prévisibles d'événements impliquant le logiciel et qui peuvent donner lieu à une situation dangereuse (voir l'ISO 14971). Il convient par conséquent d'inclure dans cette ANALYSE DE RISQUE les DANGERS qui pourraient être directement induits par le logiciel (par exemple, la fourniture d'informations propres à induire en erreur et qui pourraient donner lieu à l'administration d'un traitement inadéquat).

Toutes les ACTIVITES qui sont exigées comme faisant partie de la première catégorie des PROCESSUS sont identifiées dans le texte normatif comme « [Classes A, B, C] », indiquant ainsi qu'elles sont exigées quelle que soit la classification du logiciel auquel elles s'appliquent.

LES ACTIVITES sont exigées pour toutes les classes A, B et C pour les raisons suivantes:

– l'ACTIVITE génère un plan qui s'applique à la GESTION DES RISQUES ou à la classification de SECURITE du logiciel;

– l'ACTIVITE génère un élément de sortie qui est un élément d'entrée pour la GESTION DES RISQUES ou la classification de SECURITE du logiciel;

– l'ACTIVITE fait partie de la GESTION DES RISQUES ou de la classification de SECURITE du logiciel;

– l'ACTIVITE établit un système d'administration, de documentation ou de tenue des enregistrements qui vient à l'appui de la GESTION DES RISQUES ou de la classification de SECURITE du logiciel;

– l'ACTIVITE est en général entreprise à un moment où la classification du logiciel concernée n'est pas connue;

– l'ACTIVITE peut donner lieu à une modification qui pourrait invalider la classification actuelle de SECURITE du logiciel concernée. Ceci comprend la découverte et l'analyse de problèmes liés à la SECURITE après diffusion du logiciel.

D'autres PROCESSUS sont exigés seulement pour les SYSTEMES LOGICIELS ou LES ELEMENTS LOGICIELS classés dans les classes de SECURITE de logiciel B ou C. Les ACTIVITES exigées comme faisant partie de ce PROCESSUS sont identifiées dans le texte normatif comme « [Classe B, C] », ou « [Classe C] » indiquant ainsi qu'elles sont exigées de manière sélective en fonction de la classification du logiciel auquel elles s'appliquent.

Les ACTIVITES sont exigées sélectivement pour les logiciels de classe B ou C pour les raisons suivantes:

– l'ACTIVITE améliore la fiabilité du logiciel en exigeant plus de détails ou plus de rigueur de conception, d'essai ou autre vérification;
– l'ACTIVITE est une ACTIVITE administrative qui vient à l'appui d'une autre ACTIVITE exigée pour les classes B ou C;
– l'ACTIVITE prend en charge la correction de problèmes relatifs à la SECURITE;
– l'ACTIVITE produit des enregistrements de la conception, de la mise en œuvre, de la VERIFICATION et de la diffusion de logiciels relatifs à la SECURITE.

Les ACTIVITES sont exigées sélectivement pour les logiciels de classe C pour les raisons suivantes:

– L'ACTIVITE apporte une amélioration supplémentaire à la fiabilité du logiciel en exigeant plus de détails ou plus de rigueur ou plus d'attention à des points spécifiques de la conception, des essais ou autre VERIFICATION

Il est à noter que tous les PROCESSUS et ACTIVITES définis dans la présente norme sont considérés déterminants pour assurer le développement et la maintenance de logiciels de grande qualité. L'omission de nombre de ces PROCESSUS et ACTIVITES en tant qu'exigences pour les logiciels de classe A qui ne peuvent par définition donner lieu à un PHENOMENE DANGEREUX ne signifie pas que ces PROCESSUS et ACTIVITES ne seraient pas importants ou qu'ils ne sont pas recommandés. Leur omission permet de reconnaître que la SECURITE et l'efficacité de logiciels qui ne peuvent donner lieu à des DANGERS peuvent être facilement assurées, principalement par une ACTIVITE de validation globale lors de la conception du DISPOSITIF MEDICAL (qui est hors du domaine d'application de la présente norme) ainsi que par de simples contrôles du cycle de vie du logiciel.

A.2 Récapitulatif des exigences par classe

Le Tableau A.1 résume les classes de sécurité de logiciel qui sont attribuées à chaque exigence. Ce tableau est informatif et fourni uniquement pour commodité. La section normative identifie les classes de sécurité de logiciel pour chaque exigence.

Tableau A.1 – Récapitulatif des exigences par classe de sécurité de logiciel

Articles et paragraphes		Classe A	Classe B	Classe C
Article 4	Toutes les exigences	X	X	X
5.1	5.1.1, 5.1.2, 5.1.3, 5.1.6, 5.1.7, 5.1.8, 5.1.9	X	X	X
	5.1.5, 5.1.10, 5.1.11		X	X
	5.1.4			X
5.2	5.2.1, 5.2.2, 5.2.4, 5.2.5, 5.2.6	X	X	X
	5.2.3		X	X
5.3	5.3.1, 5.3.2, 5.3.3, 5.3.4, 5.3.6		X	X
	5.3.5			X
5.4	5.4.1		X	X
	5.4.2, 5.4.3, 5.4.4			X
5.5	5.5.1	X	X	X
	5.5.2, 5.5.3, 5.5.5		X	X
	5.5.4			X
5.6	Toutes les exigences		X	X
5.7	Toutes les exigences		X	X
5.8	5.8.4	X	X	X
	5.8.1, 5.8.2, 5.8.3, 5.8.5, 5.8.6, 5.8.7, 5.8.8		X	X
6.1	6.1	X	X	X
6.2	6.2.1, 6.2.2, 6.2.4, 6.2.5	X	X	X
	6.2.3		X	X
6.3	Toutes les exigences	X	X	X
7.1	Toutes les exigences		X	X
7.2	Toutes les exigences		X	X
7.3	Toutes les exigences		X	X
7.4	7.4.1	X	X	X
	7.4.2, 7.4.3		X	X
Article 8	Toutes les exigences	X	X	X
Article 9	Toutes les exigences	X	X	X

Annexe B
(informative)

Lignes directrices relatives aux dispositions de la présente norme

B.1 Domaine d'application

B.1.1 Objet

L'objet de la présente norme est de fournir un PROCESSUS de développement qui produira de manière homogène des LOGICIELS DE DISPOSITIFS MEDICAUX de grande qualité et sûrs. Pour cela, la présente norme identifie les ACTIVITES et TACHES minimales qui peuvent être réalisées pour acquérir la certitude que le logiciel a été développé d'une manière qui produira vraisemblablement des PRODUITS LOGICIELS hautement fiables et sûrs.

La présente annexe fournit des lignes directrices pour l'application des exigences de la présente norme. Elle n'ajoute aucune information supplémentaire ni ne modifie les exigences de la présente norme. Cette annexe peut être utilisée pour mieux comprendre les exigences de la présente norme.

Il est à noter que dans la présente norme, les ACTIVITES sont définies dans des paragraphes référencés dans les PROCESSUS et que les TACHES sont définies dans les ACTIVITES correspondantes. Par exemple, les ACTIVITES définies pour le PROCESSUS de développement du logiciel sont la planification du développement du LOGICIEL, l'analyse des exigences du logiciel, la conception ARCHITECTURALE du logiciel, la conception détaillée du logiciel, le codage du logiciel, la mise en œuvre et la VERIFICATION de L'UNITE LOGICIELLE, l'intégration et les essais d'intégration du logiciel, les essais du SYSTEME LOGICIEL et la diffusion du logiciel. Les TACHES dans ces ACTIVITEs sont les exigences individuelles.

La présente norme n'exige pas un MODELE particulier de CYCLE DE VIE DE DEVELOPPEMENT DU LOGICIEL. Cependant, la conformité à la présente norme implique effectivement des dépendances entre PROCESSUS, car les éléments d'entrée d'un PROCESSUS donné sont générés par un autre PROCESSUS. Il convient par exemple, de compléter la classification de la sécurité du logiciel du SYSTEME LOGICIEL une fois que l'ANALYSE DU RISQUE a établi les dommages qui pourraient résulter de la défaillance du SYSTEME LOGICIEL.

Du fait de ces dépendances logiques entre PROCESSUS, il est plus facile de décrire les PROCESSUS dans la présente norme comme une séquence, ce qui implique un modèle de cycle de vie de type «en cascade» ou «à passage unique». Cependant, d'autres cycles de vie peuvent également être utilisés. Certaines stratégies (modèles) de développement, qui sont définies dans l'ISO/CEI 12207 [9], comportent les stratégies suivantes (voir également le Tableau B.1):

– En cascade: La stratégie à «passage unique» également appelée «en cascade», consiste à réaliser le PROCESSUS de développement en une seule fois. De manière plus simple: on détermine les besoins du client, on définit les exigences, on conçoit le SYSTEME, on met en œuvre le SYSTEME, on le soumet aux essais, on le corrige et on le livre.
– Incrémentielle: La stratégie «incrémentielle» consiste à déterminer les besoins du client et à définir les exigences du SYSTEME puis à entreprendre le reste du développement en une séquence d'éléments de construction. Le premier élément de construction intègre une partie des capacités prévues, l'élément de construction suivant ajoute des capacités supplémentaires et ainsi de suite jusqu'à ce que le SYSTEME soit complet.
– Evolutive: La stratégie «évolutive» consiste également à développer un SYSTEME en éléments de construction mais diffère de la stratégie incrémentielle du fait qu'elle considère de prime abord que le besoin de l'utilisateur n'est pas pleinement compris et que toutes les exigences ne peuvent être définies en amont. Dans cette stratégie, les besoins du client et les exigences du SYSTEME sont partiellement définis en amont mais sont ensuite définis à chaque élément de construction suivant.

Tableau B.1 – Stratégies (modèle) de développement telles que définies dans l'ISO/CEI 12207

Stratégie de développement	Définition a priori de toutes les exigences?	Cycles de développement multiples ?	Diffusion d'un logiciel intermédiaire ?
En cascade (Passage unique)	Oui	Non	Non
Incrémentielle (Amélioration du produit pré-planifiée)	Oui	Oui	Probablement
Evolutive	Non	Oui	Oui

Quel que soit le cycle de vie choisi, il est nécessaire de maintenir les dépendances logiques entre éléments de sortie du PROCESSUS tels que les spécifications, les documents de conception et les logiciels. Le modèle de cycle de vie en cascade atteint cet objectif en retardant le début du PROCESSUS jusqu'à ce que les éléments d'entrée de ce processus soient terminés et approuvés.

D'autres cycles de vie, notamment les cycles de vie évolutifs, permettent la production des éléments de sortie du PROCESSUS avant disponibilité de tous les éléments d'entrée de ce PROCESSUS. Il est possible par exemple de spécifier un nouvel ELEMENT LOGICIEL, de le classer, de le mettre en œuvre et de le vérifier avant que l'ensemble de l'ARCHITECTURE DU LOGICIEL n'ait été finalisé. Le RISQUE encouru par de tels cycles de vie est qu'une modification ou un développement apporté à un élément de sortie d'un PROCESSUS donné invalidera l'élément de sortie d'un autre PROCESSUS. Par conséquent, tous les cycles de vie utilisent un système exhaustif de gestion de la configuration pour s'assurer que tous les éléments de sortie d'un PROCESSUS sont amenés à un état homogène et que les dépendances sont bien maintenues.

Quel que soit le cycle de vie de développement du logiciel utilisé, les principes suivants sont d'une importance primordiale:

– il convient de maintenir en un état cohérent tous les éléments de sortie de PROCESSUS; lorsqu'un élément de sortie de PROCESSUS est créé ou modifié, il convient de mettre à jour rapidement tous les éléments de sortie de PROCESSUS correspondants afin d'assurer leur cohérence les uns vis-à-vis des autres et de maintenir explicitement ou implicitement les dépendances exigées par la présente norme;
– il convient que tous les éléments de sortie des PROCESSUS soient disponibles lorsque cela est nécessaire en tant qu'éléments d'entrée pour poursuivre les travaux sur le logiciel.
– avant diffusion d'un LOGICIEL DE DISPOSITIF MEDICAL, il convient d'assurer la cohérence des éléments de sortie des PROCESSUS et d'observer toutes les dépendances entre éléments de sortie de PROCESSUS explicitement ou implicitement exigées par la présente norme.

B.1.2 Domaine d'application

La présente norme s'applique au développement et à la maintenance de LOGICIELS DE DISPOSITIFS MEDICAUX ainsi qu'au développement et à la maintenance d'un DISPOSITIF MEDICAL qui comprend un logiciel SOUP.

L'utilisation de la présente norme exige que le FABRICANT applique une GESTION DES RISQUES du DISPOSITIF MEDICAL qui soit conforme à l'ISO 14971. Par conséquent, lorsque l'ARCHITECTURE du système de DISPOSITIF MEDICAL comprend un composant acquis (il pourrait s'agir d'un composant acheté ou d'un composant de provenance inconnue), tel qu'une imprimante ou un traceur qui comprend un logiciel SOUP, ce composant acquis passe sous la responsabilité du FABRICANT et doit être inclus dans la GESTION DES RISQUES du DISPOSITIF MEDICAL. On suppose que par la réalisation correcte de la GESTION DES RISQUES pour le DISPOSITIF MEDICAL, le FABRICANT comprendrait le composant et reconnaîtrait qu'il comprend un logiciel SOUP. Le FABRICANT utilisant la présente norme ferait alors appel au PROCESSUS de

GESTION DES RISQUES du logiciel dans le cadre du PROCESSUS DE GESTION DES RISQUES du DISPOSITIF MEDICAL.

La maintenance du LOGICIEL de DISPOSITIF MEDICAL diffusé s'applique au retour d'expérience post-production acquise avec le LOGICIEL de DISPOSITIF MEDICAL. La maintenance du logiciel inclut la combinaison de toutes les actions techniques et administratives, y compris les opérations de surveillance, pour agir sur un rapport de problème afin de maintenir ou de remettre un élément dans un état lui permettant d'accomplir une fonction requise, aussi bien que des DEMANDES DE MODIFICATION liées à des PRODUITS LOGICIELS diffusés. Par exemple, ceci inclut la rectification d'un problème, le rapport réglementaire, une nouvelle validation et l'action préventive. Voir ISO/CEI 14764 [10].

B.2 Références normatives

La norme ISO/CEI 90003 [11] fournit des lignes directrices pour l'application d'un système de management de la qualité au développement du logiciel. Ces lignes directrices ne sont pas exigées par la présente norme mais hautement recommandées.

B.3 Termes et définitions

Dans toute la mesure du possible, des termes ont été spécifiés sur la base de définitions extraites de normes internationales.

La présente norme a choisi d'utiliser trois termes pour décrire la décomposition d'un SYSTEME LOGICIEL (niveau le plus élevé). Le SYSTEME LOGICIEL peut être un sous-système du DISPOSITIF MEDICAL (voir la CEI 60601-1-4 [2]) ou un DISPOSITIF MEDICAL à part entière. Le niveau inférieur qui ne peut être à nouveau décomposé à des fins d'essai ou de gestion de la configuration de logiciel est l'UNITE LOGICIELLE. Tous les niveaux de décomposition, y compris les niveaux supérieur et inférieur, peuvent être appelés ELEMENTS LOGICIELS. Un SYSTEME LOGICIEL est ainsi constitué de un ou plusieurs ELEMENTS LOGICIELS et chaque ELEMENT LOGICIEL est constitué d'une ou de plusieurs UNITES LOGICIELLES ou d'ELEMENTS LOGICIELS décomposables. Il incombe au FABRICANT de fournir la définition et la granularité des ELEMENTS LOGICIELS et des UNITES LOGICIELLES. Si la définition de ces termes reste vague, ils pourront être appliqués aux nombreuses et diverses méthodes de développement et types de logiciels utilisés dans les DISPOSITIFS MEDICAUX.

B.4 Exigences générales

Il n'existe pas de méthode connue permettant d'assurer une SECURITE à 100 % pour tout type de logiciel.

Trois principes majeurs permettent de promouvoir la SECURITE des LOGICIELS de DISPOSITIFS MEDICAUX:

– la GESTION DES RISQUES;
– le management de la qualité;
– l'ingénierie logicielle.

Pour le développement et la maintenance de logiciels sûrs de dispositifs médicaux, il est nécessaire de mettre en place une GESTION DES RISQUES comme partie intégrante d'un système de management de la qualité et utilisée comme cadre global pour l'application de méthodes et techniques appropriées d'ingénierie logicielle. La combinaison de ces trois concepts permet à un FABRICANT de DISPOSITIF MEDICAL de suivre un PROCESSUS décisionnel clairement structuré et reproductible de manière homogène pour promouvoir la SECURITE du LOGICIEL DE DISPOSITIF MEDICAL.

B.4.1 Système de management de la qualité

Un ensemble discipliné et efficace de PROCESSUS logiciels comprend des PROCESSUS organisationnels tels que la gestion, l'infrastructure, l'amélioration et la formation. Ces PROCESSUS ont été omis de la présente norme pour éviter les duplications et pour se concentrer sur l'ingénierie logicielle. Ces PROCESSUS font l'objet d'un système de management de la qualité. L'ISO 13485 [7] est une Norme Internationale qui est spécifiquement dédiée à l'application des concepts de l'assurance qualité aux DISPOSITIFS MEDICAUX. La conformité aux exigences du système de management de la qualité de l'ISO 13485 ne constitue pas automatiquement une conformité aux exigences réglementaires, nationales ou régionales. Il incombe au FABRICANT d'identifier et d'établir la conformité aux exigences réglementaires applicables.

B.4.2 GESTION DES RISQUES

Le développement des logiciels participe suffisamment aux ACTIVITES de GESTION DES RISQUES pour assurer que tous les RISQUES raisonnablement prévisibles, associés au LOGICIEL DE DISPOSITIF MEDICAL sont effectivement pris en compte.

Plutôt que de tenter de définir un PROCESSUS de GESTION DES RISQUES approprié dans cette norme relative à l'ingénierie logicielle, il est exigé que le FABRICANT applique un PROCESSUS DE GESTION DES RISQUES qui soit conforme à l'ISO 14971, dans la mesure où cette dernière traite explicitement de la GESTION DES RISQUES pour les DISPOSITIFS MEDICAUX. Les ACTIVITES de la GESTION DES RISQUES spécifiques au logiciel résultant de DANGERS dont le logiciel est une des causes sont identifiées dans un PROCESSUS explicatif décrit dans l'Article 7.

B.4.3 Classification de sécurité du logiciel

Le RISQUE associé au logiciel en tant que partie constituante d'un DISPOSITIF MEDICAL, en tant qu'accessoire d'un dispositif MEDICAL, ou en tant que DISPOSITIF MEDICAL à part entière, est utilisé comme l'élément d'entrée d'un plan de classification de la sécurité logicielle qui permet alors de déterminer les PROCESSUS à utiliser au cours du développement et de la maintenance du logiciel.

Le RISQUE est défini comme étant une combinaison de la gravité du DOMMAGE et de la probabilité de sa survenance. Cependant, il n'existe pas de consensus quant à la manière de déterminer la probabilité des défaillances logicielles sur la base de méthodes statistiques conventionnelles. Par conséquent, dans la présente norme, la classification du SYSTEME LOGICIEL est fondée sur la gravité du DANGER résultant de la défaillance du logiciel, en supposant que la défaillance aura lieu. Les SYSTEMES LOGICIELS qui contribuent à la mise en œuvre des mesures de MAITRISE DU RISQUE sont classés en fonction de la gravité du DANGER correspondant.

Si un SYSTEME LOGICIEL est décomposé en ELEMENTS LOGICIELS, alors chaque ELEMENT LOGICIEL peut avoir sa propre classification de sécurité du logiciel.

Le RISQUE lié à la défaillance d'un ELEMENT LOGICIEL ne peut être déterminé que:

– si le rôle de l'ELEMENT LOGICIEL, en termes de fonctionnalité et d'interfaces avec d'autres ELEMENTS LOGICIELS et matériels, est défini par une ARCHITECTURE du système et une ARCHITECTURE du logiciel;

– si les modifications au SYSTEME sont maîtrisées;

– après réalisation d'une ANALYSE DE RISQUE sur l'ARCHITECTURE et application des mesures de MAITRISE DU RISQUE spécifiées.

La présente norme exige que soit entrepris le nombre minimal d'ACTIVITES qui permettront de satisfaire les conditions ci-dessus pour toutes les classes de logiciel.

L'achèvement d'une ACTIVITE d'ARCHITECTURE DU LOGICIEL est le point le plus en amont du développement où l'ensemble complet des ELEMENTS LOGICIELS est défini et que l'ACTIVITE de GESTION DES RISQUES a identifié la manière dont les ELEMENTS LOGICIELS sont liés à la SECURITE. Il s'agit par conséquent du point le plus précoce au niveau duquel les ELEMENTS LOGICIELS peuvent être définitivement classés en fonction de leur effet sur la SECURITE.

Ce point correspond au point pour lequel la MAITRISE DU RISQUE commence dans l'ISO 14971.

Avant d'atteindre ce point, le PROCESSUS de GESTION DES RISQUES identifie les mesures de MAITRISE DU RISQUE ARCHITECTURAL en ajoutant par exemple des sous-systèmes de protection ou en réduisant les possibilités de défaillance logicielles qui peuvent entraîner des DOMMAGES. Après ce point, le PROCESSUS DE GESTION DES RISQUES utilise des PROCESSUS destinés à réduire la probabilité de défaillance des ELEMENTS LOGICIELS. En d'autres termes, la classification d'un ELEMENT LOGICIEL prescrit l'application à cet élément de mesures de MAITRISE DU RISQUE basée sur les PROCESSUS.

Il est probable que les FABRICANTS considèrent qu'il est utile de classer les logiciels avant ce point pour, par exemple, s'attacher plus particulièrement à des domaines de recherche particuliers, mais il convient que cette classification soit considérée comme préliminaire et ne soit pas utilisée pour justifier l'omission des PROCESSUS.

Le plan de classification de SECURITE du logiciel ne signifie pas un alignement sur la classification des RISQUES de l'ISO 14971. Alors que le plan de l'ISO 14971 classe les RISQUES en fonction de leur gravité et de leur probabilité, le plan de classification de SECURITE du logiciel définit les SYSTEMES LOGICIELS et ELEMENTS LOGICIELS en fonction des PROCESSUS à appliquer au cours de leur développement et de leur maintenance.

Au fur et à mesure que la conception évolue, de nouveaux RISQUES pourraient apparaître. Il est par conséquent recommandé d'appliquer la GESTION DES RISQUES comme partie intégrante du PROCESSUS de développement. Ceci permet de développer une conception architecturale qui identifie un ensemble complet d'ELEMENTS LOGICIELS, y compris ceux dont il est exigé une fonctionnalité correcte pour assurer un fonctionnement en toute SECURITE et ceux qui évitent que les défauts n'aboutissent à des DOMMAGES.

Il convient que l'ARCHITECTURE DU LOGICIEL favorise la différentiation des ELEMENTS LOGICIELS qui sont requis pour un fonctionnement en toute SECURITE et qu'elle décrive les méthodes utilisées pour assurer une différentiation effective de ces ELEMENTS LOGICIELS.

Comme indiqué en B.3, la présente norme a choisi d'utiliser trois termes pour décrire la décomposition d'un SYSTEME LOGICIEL (le niveau le plus élevé).

La Figure B.1 illustre le découpage possible d'ELEMENTS LOGICIELS dans un SYSTEME LOGICIEL donné et la manière dont les classes de SECURITE du logiciel seraient appliquées au groupe d'ELEMENTS LOGICIELS de la décomposition.

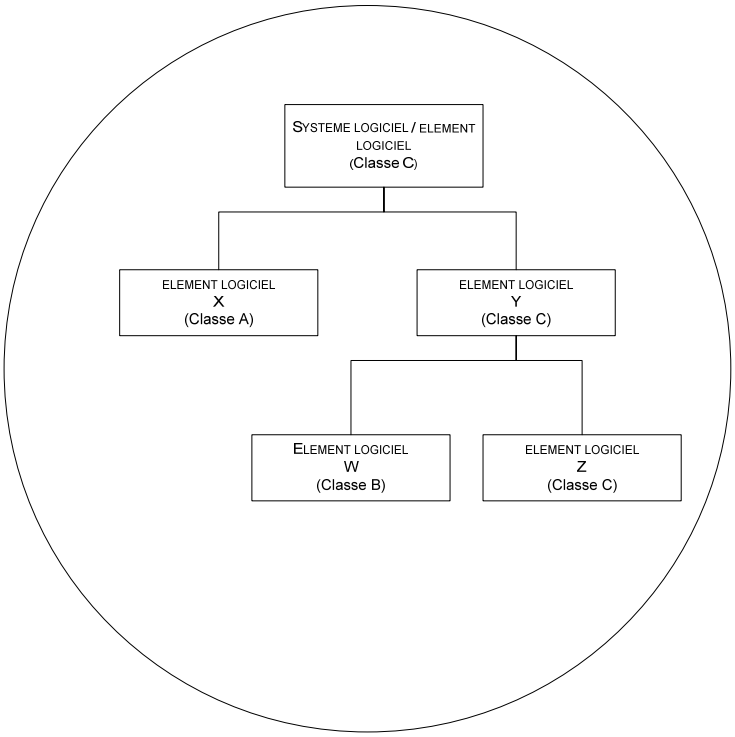

Figure B.1 – Exemple de découpage d'ELEMENTS LOGICIELS

Dans cet exemple, le FABRICANT sait, du fait du type de LOGICIEL DE DISPOSITIF MEDICAL à développer, que la classification préliminaire de SECURITE du logiciel pour le SYSTEME LOGICIEL est la classe C de SECURITE du logiciel. Pendant la conception de l'ARCHITECTURE du logiciel, le FABRICANT a décidé de découper le SYSTEME, comme illustré, en 3 ELEMENTS LOGICIELS – X, W et Z. Le FABRICANT est capable d'isoler de l'ELEMENT LOGICIEL Z toutes les contributions du SYSTEME LOGICIEL aux DANGERS qui pourraient entraîner la mort ou une blessure grave, et de l'ELEMENT LOGICIEL W toutes les contributions restantes du SYSTEME LOGICIEL aux DANGERS qui pourraient entraîner une blessure légère. L'ELEMENT LOGICIEL W est de classe B de SECURITE du logiciel et l'ELEMENT LOGICIEL Z est de classe C de SECURITE du logiciel. Par conséquent, L'ELEMENT LOGICIEL Y doit être de Classe C, conformément à 4.3 d). Le SYSTEME LOGICIEL est également, du fait de cette exigence, de classe C de SECURITE du logiciel. L'ELEMENT LOGICIEL X a été défini en classe A de SECURITE du logiciel. Le FABRICANT est capable de justifier la séparation entre les ELEMENTS LOGICIELS X et Y, ainsi qu'entre les ELEMENTS LOGICIELS W et Z, pour en assurer l'intégrité. Si la division n'est pas possible, les ELEMENTS LOGICIELS X et Y doivent être classés dans la classe C de sécurité du logiciel.

B.5 PROCESSUS de développement du logiciel

B.5.1 Planification du développement du logiciel

L'objectif de cette ACTIVITE est de planifier les TACHES de développement du logiciel de manière à réduire les RISQUES causés par le logiciel, de communiquer les procédures et les objectifs aux membres de l'équipe de développement et de s'assurer que les exigences de qualité du système pour le LOGICIEL DE DISPOSITIF MEDICAL sont remplies.

L'ACTIVITE de planification du développement du logiciel peut documenter des TACHES dans un plan unique ou dans plusieurs plans. Il est admis que certains FABRICANTS établissent des politiques et des procédures qui s'appliquent au développement de tout LOGICIEL DE DISPOSITIF MEDICAL qui leur appartient. Dans ce cas, le plan peut simplement référencer les politiques et procédures existantes. Il est admis que certains FABRICANTS élaborent un plan ou un ensemblede plans spécifique au développement de chaque PRODUIT LOGICIEL DE DISPOSITIF MEDICAL qui reprenne en détails les ACTIVITES spécifiques et fasse référence aux procédures générales. Une autre possibilité est d'adapter un plan ou un ensemble de plans au développement de chaque PRODUIT LOGICIEL DE DISPOSITIF MEDICAL. Il convient que la

planification précise les niveaux de détail nécessaires pour réaliser le PROCESSUS de développement et soit proportionnelle au RISQUE encouru. Par exemple, les SYSTEMES ou les éléments à RISQUE plus élevé pourraient faire l'objet d'un PROCESSUS de développement plus rigoureux et les TACHES pourraient être décrites de manière plus détaillée.

La planification est une ACTIVITE dynamique itérative qu'il convient de réexaminer et de remettre à jour au fur et à mesure du développement. Le plan peut évoluer pour intégrer davantage ou de meilleures informations au fur et à mesure de la compréhension du SYSTEME et du niveau d'effort nécessaire au développement du SYSTEME. Par exemple, la classification initiale de la sécurité du logiciel d'un SYSTEME peut changer suite au PROCESSUS de GESTION DES RISQUEs et du développement de l'ARCHITECTURE du logiciel. Ou encore, il peut être décidé d'intégrer UN LOGICIEL SOUP aU SYSTEME. Il est important que le(s) plan(s) soi(en)t mis à jour afin de refléter la connaissance courante du SYSTEME et le niveau de rigueur requis du SYSTEME ou des éléments du SYSTEME pour permettre de maîtriser correctement le PROCESSUS de développement.

B.5.2 Analyses des exigences du logiciel

Cette ACTIVITE exige que le FABRICANT établisse et vérifie les exigences du logiciel pour le LOGICIEL DE DISPOSITIF MEDICAL. L'établissement d'exigences vérifiables est essentiel pour déterminer ce qui doit être construit, pour déterminer que le LOGICIEL DE DISPOSITIF MEDICAL présente un comportement acceptable et pour démontrer que le LOGICIEL DE DISPOSITIF MEDICAL terminé est prêt pour utilisation. Pour démontrer que les exigences ont été mises en œuvre comme prévu, il convient que chaque exigence soit indiquée de sorte que les critères objectifs puissent être établis afin de déterminer si une mise en œuvre correcte a bien été faite. Si le PROCESSUS de GESTION DES RISQUES du dispositif impose au logiciel des exigences de MAITRISE DES RISQUES IDENTIFIES, ces exigences doivent être identifiées dans les exigences du logiciel de façon à ce qu'il soit possible d'assurer une TRAÇABILITE rattachant les mesures de MAITRISE DES RISQUES aux exigences du logiciel. Il est recommandé que toutes les exigences du logiciel soient identifiées de façon à permettre de démontrer la TRAÇABILITE entre l'exigence et les essais du SYSTEME LOGICIEL. Si dans certains pays l'approbation d'un organisme de contrôle exige la conformité à des réglementations ou à des normes internationales spécifiques, il convient de documenter cette exigence de conformité dans les exigences de logiciel. Les exigences du logiciel établissent ce qui doit être mis en œuvre dans le logiciel et par conséquent, il est nécessaire de les évaluer avant de terminer l'ACTIVITE d'analyse des exigences.

Il est fréquent de confondre les besoins du client, les éléments d'entrée de conception, les exigences du logiciel, les spécifications fonctionnelles du logiciel et les spécifications de conception du logiciel. Les éléments d'entrée de la conception sont l'interprétation des besoins du client en exigences relatives au DISPOSITIF MEDICAL, formellement documentées. Les exigences du logiciel sont les spécifications formellement documentées de ce que le logiciel réalise pour répondre aux besoins du client et de satisfaire aux éléments d'entrée de la conception. Les spécifications fonctionnelles du logiciel sont souvent incluses dans les exigences du logiciel et définissent en détails ce que le logiciel réalise pour satisfaire à ses exigences même s'il existe de nombreux autres moyens de satisfaire également aux exigences. Les spécifications de conception du logiciel définissent la manière dont le logiciel sera conçu et décomposé pour mettre en œuvre ses exigences et ses spécifications fonctionnelles.

Historiquement, les exigences du logiciel, les spécifications fonctionnelles et les spécifications de conception étaient écrites sous la forme d’un ensemble composé d'un ou de plusieurs documents. Il est aujourd'hui possible de considérer ces informations comme des éléments de données au sein d'une base de données commune. Chaque élément aurait alors un ou plusieurs attributs qui définiraient son objectif et ses liens avec d'autres éléments dans la base de données. Cette approche permet de présenter et d'imprimer différentes vues des informations qui correspondent le mieux à chaque ensemble d'utilisateurs destinataires

(par exemple, marketing, FABRICANTS, contrôleurs, auditeurs); cette approche prend également en charge la TRAÇABILITE ce qui permet de démontrer que la mise en œuvre a été correctement réalisée et dans quelle mesure les jeux d'essais permettent de vérifier les exigences. Les outils qui prennent en charge cette approche peuvent être tout simplement des documents hypertext utilisant des hyperliens HTML ou aussi complexes et fonctionnels que des outils d'ingénierie logicielle assistée par ordinateur (en anglais, CASE Computer Aided Software Engineering).

Le PROCESSUS d'exigences SYSTEME n'est pas l'objet de la présente norme. Cependant, la décision de mettre en œuvre avec le logiciel une fonctionnalité DE DISPOSITIF MEDICAL est normalement prise lors de la conception du SYSTEME. Certaines ou toutes les exigences du SYSTEME font l'objet d'une affectation pour être mises en œuvre dans le logiciel. L'ACTIVITE d'analyse des exigences du logiciel consiste à analyser les exigences attribuées au logiciel par le PROCESSUS des exigences SYSTEME et à en extraire un ensemble exhaustif d'exigences logicielles qui reflètent les exigences qui lui sont attribuées.

Pour assurer l'intégrité du SYSTEME, il convient que le FABRICANT prévoit un mécanisme de prise en charge des modifications et clarifications apportées aux exigences du SYSTEME afin de corriger les impossibilités, les incohérences ou les ambiguïtés, que ce soit dans les exigences du SYSTEME d'origine ou celles du logiciel.

Le PROCESSUS de collecte et d'analyse des exigences du SYSTEME et du logiciel peut être itératif. La présente norme n'exige pas que les PROCESSUS soient séparés en deux couches strictement délimitées. Dans la pratique, l'ARCHITECTURE DU SYSTEME et l'ARCHITECTURE du logiciel sont souvent décrites simultanément et leurs exigences respectives sont ensuite consignées par écrit sous forme de couche.

B.5.3 Conception ARCHITECTURALE du logiciel

Cette ACTIVITE exige que le FABRICANT définisse les principaux composants structurels du logiciel, leurs caractéristiques visibles de l'extérieur, ainsi que les relations qui existent entre eux. Si le comportement d'un composant peut affecter d'autres composants, il convient que ce comportement soit décrit dans l'ARCHITECTURE logicielle. Cette description est notamment importante lorsque le comportement peut affecter des composants du DISPOSITIF MEDICAL extérieurs au logiciel. Les décisions relatives à l'ARCHITECTURE sont extrêmement importantes pour la mise en œuvre des mesures de MAITRISE DU RISQUE. Si on ne comprend pas (et si l'on ne documente pas) le comportement d'un composant qui peut affecter d'autres composants, il sera presque impossible de démontrer que le SYSTEME est sûr. Une ARCHITECTURE logicielle est nécessaire pour s'assurer de la mise en œuvre correcte des exigences du logiciel. L'ARCHITECTURE du logiciel n'est complète que si toutes les exigences logicielles peuvent être mises en œuvre par les ELEMENTS LOGICIELS identifiés. La conception et la mise en œuvre du logiciel étant dépendantes de l'ARCHITECTURE, l'ARCHITECTURE est vérifiée pour terminer cette ACTIVITE. Une EVALUATION technique est généralement utilisée pour la VERIFICATION de l'ARCHITECTURE.

La classification des ELEMENTS LOGICIELS pendant l'ACTIVITE d'ARCHITECTURE du logiciel génère une base qui permet par la suite de choisir des PROCESSUS logiciels. Les enregistrements de cette classification sont mis en MAITRISE DES MODIFICATIONS comme partie intégrante du DOSSIER DE GESTION DES RISQUES.

La classification peut être invalidée par de nombreux événements ultérieurs, tels que, par exemple:

– les modifications de la spécification du SYSTEME, de la spécification ou de l'ARCHITECTURE du logiciel;
– la découverte d'erreurs dans l'ANALYSE DU RISQUE, notamment en ce qui concerne des DANGERS non prévus; et
– la découverte de l'infaisabilité d'une exigence, notamment une mesure de MAITRISE DU RISQUE.

Par conséquent, pendant toutes les ACTIVITES faisant suite à la conception de l'ARCHITECTURE du logiciel, il convient de réévaluer la classification du SYSTEME LOGICIEL et des ELEMENTS LOGICIELS et peut-être de la réviser. Ceci déclencherait une reprise du travail POUR L'APPLICATION DE PROCESSUS supplémentaires à un ELEMENT LOGICIEL suite à son passage dans une classe supérieure. Le PROCESSUS de gestion de la configuration du logiciel (Article 8) est utilisé pour s'assurer que toute retouche nécessaire a été identifiée et réalisée.

B.5.4 Conception détaillée du logiciel

Cette ACTIVITE exige que le FABRICANT décompose les ELEMENTS LOGICIELS et interfaces définis dans l'ARCHITECTURE pour créer des UNITES LOGICIELLES et leurs interfaces. Même si les UNITES LOGICIELLES sont souvent considérées comme étant une fonction ou un module simple, cette opinion n'est pas toujours valable. Nous avons défini l'UNITE LOGICIELLE comme étant un ELEMENT LOGICIEL qui n'est pas subdivisé en éléments plus petits. Les UNITES LOGICIELLES peuvent être soumises à des essais séparément. Il convient que le FABRICANT définisse le niveau de détails de l'UNITE LOGICIELLE. La conception détaillée spécifie des algorithmes, des représentations des données, des interfaces entre les différentes UNITES LOGICIELLES et les interfaces entre les UNITES LOGICIELLES et les structures de données. La conception détaillée doit également traiter du conditionnement (programmation) du PRODUIT LOGICIEL. Il est nécessaire de documenter la conception de chaque UNITE LOGICIELLE et de ses interfaces de manière à pouvoir correctement la mettre en œuvre. La conception détaillée renseigne les détails nécessaires à la construction du logiciel. Il convient qu'elle soit suffisamment complète pour que le programmateur n'ait pas à prendre des décisions de conception circonstancielles.

Un ELEMENT LOGICIEL peut être décomposé de sorte que seul un nombre infime de nouveaux ELEMENTS LOGICIELS mette en œuvre l'exigence relative à la SECURITE de l'ELEMENT LOGICIEL d'origine. Les autres ELEMENTS LOGICIELS ne mettent pas en œuvre des fonctions relatives à la SECURITE et peuvent être reclassés dans une classe inférieure de sécurité logicielle. Cependant, cette décision fait en elle-même partie du PROCESSUS DE GESTION DES RISQUES et elle est consignée dans le DOSSIER DE GESTION DES RISQUES.

Etant donné qu'une mise en œuvre dépend d'une conception détaillée, il est nécessaire de vérifier la conception détaillée avant de terminer l'ACTIVITE. Une EVALUATION technique est généralement effectuée pour la VERIFICATION de la conception détaillée. Le paragraphe 5.4.4 exige que le fabricant vérifie les éléments de sortie des activités de conception détaillée. La conception spécifie la manière dont les exigences doivent être mises en œuvre. Si la conception comporte des défauts, le code ne mettra pas correctement les exigences en application.

Il convient que le fabricant vérifie les caractéristiques de la conception qu'il estime importantes pour la SECURITE, s'il en existe. Les exemples de ces caractéristiques comprennent:

– la mise en œuvre des évènements prévus, les entrées, les sorties, les interfaces, le logigramme, l'allocation du processeur principal (CPU), l'allocation des ressources mémoire, les définitions de l'erreur et de l'exception, l'isolement de l'erreur et de l'exception, et la reprise sur erreur;

– la définition de l'état défectueux dans lequel toutes les fautes qui peuvent entraîner une situation dangereuse sont traitées, avec les évènements et les transitions;

– l'initialisation des variables, la gestion de la mémoire; et

– les réinitialisations à chaud et à froid, la mis en veille et les autres changements d'état qui peuvent affecter les mesures de MAITRISE DU RISQUE.

B.5.5 Mise en œuvre et vérification de l'UNITE LOGICIELLE

Cette ACTIVITE exige que le FABRICANT écrive et vérifie le code des UNITES LOGICIELLES. La conception détaillée doit être traduite en code source. Le codage représente le point où s'achève la décomposition des spécifications et où commence la composition du logiciel exécutable. Pour assurer la cohérence des caractéristiques souhaitables du code, il convient

d'utiliser des normes de codage afin de prescrire un style de codage préférentiel. Les normes de codage comprennent par exemple, des exigences d'intelligibilité, des règles ou des restrictions d'usage du langage et une gestion de la complexité. Pour chaque unité, le code est vérifié afin de s'assurer qu'il fonctionne comme spécifié dans la conception détaillée et qu'il est conforme aux normes de codage spécifiées.

Le paragraphe 5.5.5 exige que le FABRICANT vérifie le code. Si le code ne met pas correctement en application la conception, les performances du logiciel de DISPOSITIF MEDICAL ne seront pas celles attendues.

B.5.6 Intégration et essai d'intégration du logiciel

Cette ACTIVITE exige que le FABRICANT planifie et exécute l'intégration des UNITES LOGICIELLES dans des ELEMENTS LOGICIELS d'ensemble ainsi que l'intégration des ELEMENTS LOGICIELS dans des ELEMENTS LOGICIELS d'ensemble plus élevés, et qu'il vérifie que les ELEMENTS LOGICIELS qui en résultent se comportent comme prévu.

Les approches correspondantes peuvent aller d'une intégration non incrémentielle à toute forme d'intégration incrémentielle. Les caractéristiques de l'ELEMENT LOGICIEL à assembler imposent la méthode d'intégration choisie.

Les essais d'intégration de logiciel portent sur le transfert des données et leur contrôle entre les interfaces internes et externes d'un ELEMENT LOGICIEL. Les interfaces externes sont celles que l'ELEMENT LOGICIEL partage avec d'autres logiciels, y compris le logiciel du système d'exploitation ainsi qu'avec le matériel du DISPOSITIF MEDICAL.

Il convient que la rigueur des essais d'intégration et le niveau de détail de la documentation associée aux essais d'intégration soient à la mesure du RISQUE associé au dispositif, qu'ils correspondent à la dépendance du dispositif vis-à-vis du logiciel pour les fonctions potentiellement DANGEREUSES et qu'ils tiennent compte du rôle que jouent les ELEMENTS LOGICIELS spécifiques dans des fonctions à haut RISQUE du DISPOSITIF MEDICAL. Par exemple, même s'il est recommandé de soumettre aux essais tous les ELEMENTS LOGICIELS, il convient de soumettre ceux qui ont un effet sur la SECURITE à des essais plus directs, plus approfondis et plus détaillés.

Le cas échéant, les essais d'intégration démontrent le comportement du programme aux limites de ses domaines d'entrée et de sortie et confirment les réponses du programme à des éléments d'entrée non valables, inattendus et particuliers. Les actions du programme sont révélées lorsqu'elles reçoivent des combinaisons d'éléments d'entrée ou des séquences d'éléments d'entrée inattendues ou lorsque des exigences de temporisation bien définies sont violées. Si nécessaire, il convient d'inclure dans les exigences d'essai du plan, les essais de type «boîte blanche» à réaliser dans le cadre des essais d'intégration.

Les essais de «boîte blanche», également appelés *essais transparents*, *structurels, de «boîte claire» et de «boîte ouverte»* sont une technique d'essai pour sélectionner les données d'essai qui utilise une connaissance explicite du fonctionnement interne de l'ELEMENT LOGICIEL soumis aux essais. Les essais de «boîte blanche» utilisent une connaissance spécifique de l'ELEMENT LOGICIEL pour examiner ces résultats en sortie. L'essai n'est précis que si le contrôleur sait ce qu'est supposé faire l'ELEMENT LOGICIEL. Le contrôleur peut alors voir si l'ELEMENT LOGICIEL déroge à son objectif prévu. Les essais de «boîte blanche» ne peuvent garantir que la spécification dans son ensemble a été mise en œuvre car ils se concentrent sur des essais de mise en œuvre de l'ELEMENT LOGICIEL. Les essais de «boîte noire» également appelés essais comportementaux, fonctionnels, de «boîte opaque» et de «boîte fermée» vérifient la spécification fonctionnelle et il n'est pas possible de garantir que toutes les parties de la mise en œuvre ont été contrôlées. Ainsi, les essais de «boîte noire» ont pour base la spécification et décèleront des omissions indiquant que telle partie de la spécification n'a pas été remplie. Les essais de «boîte blanche» se fondent sur la mise en œuvre et décèleront des commandes indiquant que telle partie de la mise en œuvre est défectueuse. Pour contrôler pleinement un PRODUIT LOGICIEL, des essais de «boîte noire» et des essais de «boîte blanche» pourraient être exigés.

Les plans et la documentation d'essai identifiés en 5.6 et 5.7 peuvent être des documents individuels liés à des phases spécifiques de développement ou de prototypes évolutifs. Ils pourraient également être associés de telle sorte qu'un seul document ou ensemble de documents couvre les exigences de sous-sections multiples. Tout ou partie de ces documents pourrait être incorporé dans des documents de projet de niveau supérieur, tels que le plan d'assurance qualité du logiciel ou du projet ou un plan d'essai global qui traite de tous les aspects des essais du matériel et du logiciel. Dans ce cas, il convient de créer un renvoi qui identifie la manière dont les divers documents de projet sont corrélés à chacune des TACHES d'intégration du logiciel.

Les essais d'intégration du logiciel peuvent être réalisés dans un environnement simulé, sur un matériel cible réel ou sur le DISPOSITIF MEDICAL complet.

Le paragraphe 5.6.2 exige que le FABRICANT vérifie les éléments de sortie de l'ACTIVITE d'intégration du logiciel. Le résultat de l'ACTIVITE d'intégration de logiciel est représenté par les ELEMENTS LOGICIELS intégrés. Ces ELEMENTS LOGICIELS intégrés doivent fonctionner correctement pour que l'ensemble du LOGICIEL DE DISPOSITIF MEDICAL fonctionne correctement et en toute SECURITE.

B.5.7 Essais du SYSTEME LOGICIEL

Cette ACTIVITE exige que le FABRICANT vérifie la fonctionnalité du logiciel en s'assurant que les exigences correspondantes ont été mises en œuvre avec succès.

Les essais du SYSTEME LOGICIEL démontrent que la fonctionnalité spécifiée existe. Ces essais vérifient la fonctionnalité et les performances du programme tel que construit, par rapport aux exigences du logiciel.

Les essais du SYSTEME LOGICIEL portent sur les essais fonctionnels (boîte noire), bien qu'il pourrait être souhaitable d'utiliser des méthodes «boîte blanche» (voir paragraphe précédent) pour plus d'efficacité de certains essais, en initiant des conditions de contraintes ou des défauts, ou en augmentant la portée des essais de qualification du code. L'organisation des essais par types et étapes est flexible, il convient cependant de démontrer et documenter la couverture des exigences de la MAITRISE DU RISQUE, de l'aptitude à l'usage et des types d'essais (par exemple, défauts, installations, contraintes).

Les essais du SYSTEME LOGICIEL vérifient le logiciel intégré et peuvent être effectués dans un environnement simulé, sur un matériel cible réel ou sur le DISPOSITIF MEDICAL complet.

Lorsqu'une modification est apportée à un SYSTEME LOGICIEL (même si elle est infime), il convient de déterminer le niveau des essais de régression (et non uniquement les essais de la modification particulière) pour s'assurer qu'aucun effet secondaire imprévu n'a été introduit. Il convient de planifier et de documenter ces essais de régression (en indiquant les raisons pour lesquelles les essais du SYSTEME LOGICIEL n'ont pas été totalement recommencés).

Les responsabilités relatives aux essais du SYSTEME LOGICIEL peuvent être dispersées en plusieurs lieux et confiées à différentes organisations. Cependant, quelle que soit la répartition des TACHES, les rapports contractuels, les sources de composants ou l'environnement de développement, le FABRICANT du dispositif conserve la responsabilité finale et doit s'assurer que le logiciel fonctionne correctement pour son usage prévu.

Si des ANOMALIEs non décelées au cours des essais sont récurrentes mais qu'il a été décidé de ne pas les corriger, il est nécessaire d'EVALUER ces ANOMALIES au titre de l'analyse des DANGERS afin de vérifier qu'elles n'affectent pas la SECURITE du dispositif. Il convient de comprendre les causes inhérentes et les symptômes des ANOMALIES et de documenter les raisons pour lesquelles elles n'ont pas été corrigées.

Le paragraphe 5.7.4 exige que les résultats des essais du SYSTEME LOGICIEL soient EVALUES pour s'assurer que les résultats prévus sont bien obtenus.

B.5.8 Diffusion du logiciel

Cette ACTIVITE exige que le FABRICANT documente la VERSION diffusée du LOGICIEL DE DISPOSITIF MEDICAL, qu'il précise comment elle a été créée et qu'il se conforme à des procédures appropriées de diffusion du logiciel.

Il convient que le FABRICANT soit capable de démontrer que le logiciel qui a été développé sur la base du PROCESSUS de développement est bien le logiciel qui est diffusé. Si nécessaire, il convient que le FABRICANT soit capable de récupérer le logiciel et les outils utilisés pour le générer et il est recommandé que l'entreposage, le conditionnement et la livraison du logiciel réduisent au minimum les RISQUES de DOMMAGES ou d'usage abusif. Il convient d'établir des procédures bien définies pour s'assurer que ces TACHES sont réalisées correctement et qu'elles donnent des résultats cohérents.

B.6 PROCESSUS de maintenance du logiciel

B.6.1 Etablissement du plan de maintenance du logiciel

LE PROCESSUS de maintenance du logiciel diffère, par rapport au PROCESSUS de développement du logiciel, de deux manières:

– Le FABRICANT est autorisé à utiliser un PROCESSUS moins étendu que le PROCESSUS de développement complet du logiciel pour mettre en œuvre des modifications rapides en réponse à des problèmes urgents.

– Lorsqu'il répond au RAPPORT DE PROBLEMES du logiciel, concernant le produit diffusé, le FABRICANT traite non seulement des problèmes mais doit également satisfaire aux réglementations locales (en général, en mettant en place un plan de surveillance proactif pour le recueil des données du problème sur le terrain et pour communiquer avec les utilisateurs et les autorités réglementaires au sujet du problème).

Le paragraphe 6.1 exige que ces PROCESSUS soient définis dans un plan de maintenance.

Cette ACTIVITE exige que le FABRICANT élabore ou identifie des procédures de mise en œuvre des ACTIVITES et TACHES de maintenance. Pour la mise en œuvre des actions correctives, la MAITRISE DES MODIFICATIONS en maintenance et la gestion de la diffusion du logiciel révisé, il convient que le FABRICANT consigne et résolve les rapports de problèmes et les demandes des utilisateurs tout en gérant les modifications apportées au LOGICIEL DE DISPOSITIF MEDICAL. Ce PROCESSUS est activé lorsque le LOGICIEL DE DISPOSITIF MEDICAL subit des modifications touchant au code et à la documentation correspondante du fait d'un problème ou d'une nécessité d'amélioration ou d'adaptation. L'objectif est de modifier le LOGICIEL DE DISPOSITIF MEDICAL diffusé tout en préservant son intégrité. Ce PROCESSUS couvre également la migration du LOGICIEL DE DISPOSITIF MEDICAL vers des environnements ou des plates-formes pour lesquels il n'était pas initialement diffusé. Les ACTIVITES prévues dans cet article sont spécifiques au PROCESSUS de maintenance; cependant, le PROCESSUS de maintenance pourrait utiliser d'autres PROCESSUS de la présente norme.

Il est nécessaire que le FABRICANT planifie la manière dont les ACTIVITES et LES TACHES du PROCESSUS de maintenance seront réalisées.

B.6.2 Analyse des problèmes et des modifications

Cette ACTIVITE exige que le FABRICANT analyse le retour d'information pour son effet; qu'il vérifie les problèmes qui sont rapportés; et qu'il envisage, choisisse et obtienne l'approbation de la mise en œuvre d'une option de modification donnée. Les problèmes et les autres DEMANDES DE MODIFICATION peuvent affecter les performances, la SECURITE ou les autorisations réglementaires d'un DISPOSITIF MEDICAL. Une analyse est nécessaire pour déterminer s'il existe d'éventuels effets induits par un RAPPORT DE PROBLEME ou si ces effets nécessiteront une modification pour corriger un problème ou répondre à une demande. Il est

notamment important de vérifier, par analyse de TRAÇABILITE ou de régression, que les mesures de MAITRISE DU RISQUE intégrées au dispositif ne sont pas altérées ou modifiées de manière préjudiciable par la modification logicielle mise en œuvre dans le cadre de l'ACTIVITE de maintenance du logiciel. Il est également important de vérifier que le logiciel modifié n'entraîne pas un DANGER ou n'occulte pas un RISQUE alors que précédemment il n'entraînait pas de DANGER ou n'occultait pas de RISQUES dans le logiciel. La classification de SECURITE d'un ELEMENT LOGICIEL pourrait avoir changé si la modification du logiciel entraîne à présent un DANGER ou occulte un RISQUE.

Il est important de faire la distinction entre maintenance du logiciel (Article 6) et résolution des problèmes de logiciel (Article 9).

Le PROCESSUS de maintenance du logiciel est axé sur l'apport d'une réponse appropriée aux retours d'information après diffusion du PRODUIT LOGICIEL. Le PROCESSUS de maintenance de logiciel, en tant que partie intégrante d'un DISPOSITIF MEDICAL, doit s'assurer que:

– les RAPPORTS DE PROBLEMES relatifs à la sécurité sont traités et notifiés aux autorités compétentes de réglementation et aux utilisateurs concernés;
– les PRODUITS LOGICIELS sont revalidés et rediffusés après modification avec des contrôles formels qui permettent de s'assurer que le problème a été rectifié et que les problèmes ultérieurs sont évités;
– le FABRICANT vérifie si d'autres produits logiciels auraient pu être affectés et prend les mesures nécessaires.

La résolution des problèmes de logiciel est axée sur le fonctionnement d'un système de contrôle global qui:

- analyse LES RAPPORTS DE PROBLEME et identifie l'ensemble des implications du problème;
- décide d'effectuer un certain nombre de modifications et identifie tous leurs effets secondaires;
- met en œuvre toutes les modifications tout en assurant la cohérence des ELEMENTS DE CONFIGURATION du logiciel, y compris le DOSSIER DE GESTION DES RISQUES;
- VERIFIE la mise en œuvre des modifications.

Le PROCESSUS de maintenance du logiciel utilise le PROCESSUS de résolution des problèmes de logiciel. Le PROCESSUS de maintenance de logiciel prend les décisions de haut niveau relatives aux rapports de problèmes (s'il existe un problème, s'il a un effet significatif sur la SECURITE, quelles sont les modifications nécessaires et quand doivent-elles être mises en œuvre) et utilise le PROCESSUS de résolution des problèmes de logiciel pour analyser le RAPPORT DE PROBLEME afin d'en révéler toutes les implications et de générer d'éventuelles DEMANDES DE MODIFICATION qui identifient tous les ELEMENTS DE CONFIGURATION à modifier, ainsi que toutes les étapes de VERIFICATION nécessaires.

B.6.3 Mise en œuvre de la modification

Cette ACTIVITE exige que le FABRICANT utilise un PROCESSUS établi pour réaliser la modification. Si un PROCESSUS de maintenance n'a pas été défini, les TACHES DE PROCESSUS DE DEVELOPPEMENT appropriées peuvent être utilisées pour réaliser la modification. Il convient également que le FABRICANT s'assure que la modification n'a pas d'effet négatif sur d'autres parties du LOGICIEL DE DISPOSITIF MEDICAL. L'analyse de l'effet d'une modification sur l'ensemble du LOGICIEL de DISPOSITIF MEDICAL est nécessaire sauf si le LOGICIEL de DISPOSITIF MEDICAL est traité dans le cadre d'un nouveau développement. L'étendue des essais de régression qui seront effectués pour s'assurer que les parties du LOGICIEL de DISPOSITIF MEDICAL non modifiées se comportent comme avant la modification, doit être justifiée.

B.7 PROCESSUS de GESTION DES RISQUES du logiciel

La GESTION DES RISQUES DU LOGICIEL fait partie de la GESTION DES RISQUES du DISPOSITIF MEDICAL dans son ensemble et ne peut être traitée correctement de manière isolée. La présente norme exige l'utilisation d'un PROCESSUS de GESTION DES RISQUES conforme à l'ISO 14971. La GESTION DES RISQUES telle que définie dans l'ISO 14971 traite spécifiquement un cadre de gestion efficace des RISQUES liés à l'utilisation des DISPOSITIFS MEDICAUX. Une partie de l'ISO 14971 traite de la MAITRISE DES RISQUES identifiés, associés à chaque DANGER identifié au cours de l'ANALYSE DE RISQUE. Le PROCESSUS de GESTION DES RISQUES du logiciel défini dans la présente norme fournit des exigences supplémentaires pour LA MAITRISE DU RISQUE des logiciels, y compris ceux identifiés au cours de l'ANALYSE DU RISQUE comme contribuant potentiellement à une situation dangereuse ou ceux utilisés pour la MAITRISE DES RISQUES des dispositifs médicaux. Le PROCESSUS de GESTION DES RISQUES du logiciel est inclus dans la présente norme pour deux raisons:

a) les destinataires prévus de la présente norme ont besoin de comprendre les exigences minimales des mesures de MAITRISE DU RISQUE du logiciel dans leur domaine de compétence;

b) la norme générale de GESTION DES RISQUES l'ISO 14971, citée comme référence normative dans la présente norme, ne traite pas de manière spécifique de la MAITRISE DU RISQUE du logiciel ni du positionnement de la MAITRISE DU RISQUE dans le cycle de vie de développement du logiciel.

La GESTION DES RISQUES du logiciel fait partie de la GESTION DES RISQUES du DISPOSITIF MEDICAL dans son ensemble. Les plans, les procédures et la documentation requis pour les ACTIVITES de GESTION DES RISQUES du logiciel peuvent être une série de documents séparés ou un seul document ou peuvent encore être intégrés aux ACTIVITES et à la documentation de GESTION DES RISQUES du DISPOSITIF MEDICAL tant que toutes les exigences de la présente norme sont remplies.

B.7.1 Analyse du logiciel en termes de contribution à des situations dangereuses

Il est prévu que l'analyse des PHENOMENES DANGEREUX du dispositif identifie les situations dangereuses ainsi que les mesures de MAITRISE DU RISQUE correspondantes afin d'en réduire la probabilité et/ou la gravité à un niveau acceptable. Il est également prévu que des mesures de MAITRISE DU RISQUE seront affectées à des fonctions logicielles conçues pour mettre en œuvre ces mesures.

Cependant, il n'est pas prévu que toutes les situations dangereuses du dispositif puissent être identifiées avant que l'ARCHITECTURE du logiciel ne soit produite. A cette étape, on sait la manière dont les fonctions logicielles seront mises en œuvre dans les composants logiciels et la faisabilité des mesures de MAITRISE DU RISQUE attribuées aux fonctions logicielles peut être EVALUEE. Il convient à ce moment là de réviser l'analyse des PHENOMENES DANGEREUX des dispositifs afin d'inclure:

- les situations dangereuses révisées;
- les mesures de MAITRISE DU RISQUE et les exigences logicielles révisées;
- les nouvelles situations dangereuses résultant du logiciel, par exemple des situations dangereuses liées à des facteurs humains.

Il convient que l'ARCHITECTURE de logiciel inclut des stratégies crédibles de découpage des composants logiciels de façon à ce qu'il n'y ait pas d'interaction contraire à la SECURITE.

B.8 PROCESSUS de gestion de configuration du logiciel

Le PROCESSUS de gestion de la configuration du logiciel est un PROCESSUS appliquant des procédures administratives et techniques pendant le cycle de vie du logiciel afin d'identifier et de définir des ELEMENTS LOGICIELS, y compris la documentation, dans un SYSTEME DONNE; de maîtriser les modifications et les versions des ELEMENTS LOGICIELS; et de consigner et de rendre compte de l'état des ELEMENTS LOGICIELS et des DEMANDES DE MODIFICATION. La GESTION DE LA CONFIGURATION DU LOGICIEL est nécessaire pour recréer un ELEMENT LOGICIEL, en identifier ses parties constitutives et fournir un historique des modifications qu'il a subies.

B.8.1 Identification de la configuration

Cette ACTIVITE exige que le FABRICANT identifie de manière univoque des éléments de CONFIGURATION DU LOGICIEL et LEURS VERSIONS. Ceci est nécessaire à l'identification des ELEMENTS DE CONFIGURATION du logiciel et des versions incluses dans le LOGICIEL DE DISPOSITIF MEDICAL.

B.8.2 Maîtrise des modifications

Cette ACTIVITE exige que le FABRICANT maîtrise les modifications apportées aux ELEMENTS DE CONFIGURATION du logiciel et qu'il consigne les informations identifiant les DEMANDES DE MODIFICATION et fournissant la documentation relative à leur prise en charge. Cette ACTIVITE est nécessaire pour s'assurer que des modifications non autorisées ou involontaires ne sont pas effectuées sur des éléments de configuration du LOGICIEL et que les DEMANDES DE MODIFICATION approuvées sont pleinement mises en œuvre et vérifiées.

Les DEMANDES DE MODIFICATION peuvent être approuvées par un comité de MAITRISE DES MODIFICATIONS ou par un directeur ou un responsable technique selon le plan de gestion de la configuration du logiciel. La TRAÇABILITE des DEMANDES DE MODIFICATION approuvées est rattachée à la modification réelle et à la VERIFICATION du logiciel. L'exigence est que chaque modification réelle soit reliée à une DEMANDE DE MODIFICATION et qu'il existe une documentation montrant que la DEMANDE DE MODIFICATION a été approuvée. La documentation pourrait être les procès-verbaux de réunion du comité de maîtrise des modifications, une signature d'approbation ou un enregistrement dans une base de données.

B.8.3 Documentation relative à l'état de la configuration

Cette ACTIVITE exige que le FABRICANT conserve des enregistrements de l'historique des éléments de CONFIGURATION DU LOGICIEL. Cette ACTIVITE est nécessaire pour déterminer quand et pourquoi des modifications ont été effectuées. Un accès à ces informations est nécessaire pour s'assurer que les ELEMENTS DE CONFIGURATION du logiciel comportent uniquement des modifications autorisées.

B.9 PROCESSUS de résolution de problème logiciel

Le PROCESSUS de résolution des problèmes de logiciel est un PROCESSUS d'analyse et de résolution des problèmes (y compris les non-conformités), quelle que soit leur nature ou source, y compris ceux découverts pendant le développement, la maintenance ou autres PROCESSUS. L'objectif est de fournir en temps opportun un moyen responsable et documenté pour s'assurer que les problèmes décelés sont analysés et résolus et que les tendances sont reconnues. Ce PROCESSUS est quelquefois appelé «localisation des défauts» dans la littérature d'ingénierie logicielle. Il est appelé «résolution des problèmes» dans les normes ISO/CEI 12207 [9] et CEI 60601-1-4 [2], Amendement 1. Dans la présente norme, nous avons choisi de l'appeler «résolution des problèmes de logiciel».

Cette ACTIVITE exige que le FABRICANT utilise le PROCESSUS de résolution des problèmes de logiciel lorsqu'un problème ou une non-conformité est identifié. Cette ACTIVITE est nécessaire pour s'assurer que les problèmes décelés sont analysés et évalués en termes de pertinence éventuelle vis-à-vis de la SECURITE (comme spécifié dans l'ISO 14971).

La manière de traiter les problèmes ou les non-conformités fait l'objet de plan(s) ou de procédures de développement du logiciel, comme exigé en 5.1. Ceci implique qu'il faut spécifier à chaque stade du cycle de vie les aspects du PROCESSUS de résolution des problèmes de logiciel qui seront formels et documentés ainsi que le moment où les problèmes et non-conformités doivent être introduits dans le PROCESSUS de résolution des problèmes de logiciel.

Annexe C
(informative)

Relations avec d'autres normes

C.1 Généralités

La présente norme s'applique au développement et à la maintenance des logiciels de DISPOSITIFS MEDICAUX. Le logiciel est considéré être un sous-système du DISPOSITIF MEDICAL ou est lui-même un DISPOSITIF MEDICAL. La présente norme doit être utilisée conjointement à d'autres normes pertinentes pour le développement d'un DISPOSITIF MEDICAL.

Les normes de gestion de DISPOSITIFS MEDICAUX telles que l'ISO 13485 [7] (voir l'Article C.2 et l'Annexe D) et l'ISO 14971 (voir l'Annexe C.3) fournissent un environnement de gestion qui établit une base permettant à une organisation de développer des produits. Les normes de sécurité telles que la CEI 60601-1 [1] (voir l'Article C.4) et la CEI 61010-1 [4] (voir l'Article C.5) donnent des instructions spécifiques pour la création de DISPOSITIFS MEDICAUX sûrs. Lorsque le logiciel fait partie intégrante de ces DISPOSITIFS MEDICAUX, la CEI 62304 fournit des instructions plus détaillées quant aux exigences de développement et de maintenance de logiciels DE DISPOSITIFS MEDICAUX sûrs. De nombreuses autres normes telles que l'ISO/CEI 12207 [9] (voir l'Article C.6), la CEI 61508-3 [3] (voir l'Article C.7), et l'ISO/CEI 90003 [11] peuvent être consultées en tant que sources de méthodes, d'outils et de techniques qui peuvent être utilisés pour mettre en œuvre les exigences de la CEI 62304. La Figure C.1 illustre les liens entre ces normes.

Lorsque les articles ou les exigences d'autres normes sont cités, les termes définis dans les éléments cités sont des termes définis dans une norme autre que la présente norme.

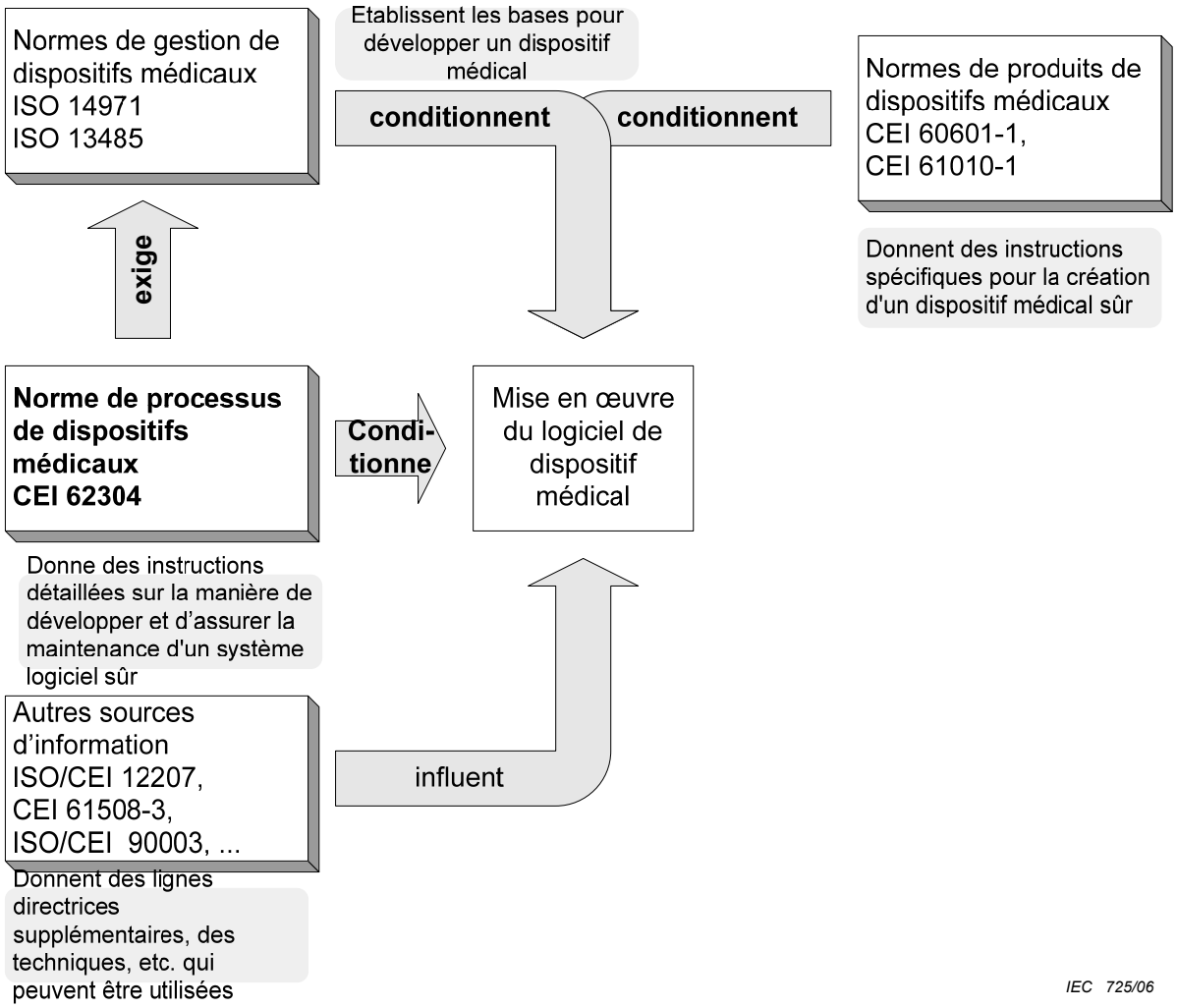

Figure C.1 – Relation des principales normes de DISPOSITIFS MEDICAUX avec la CEI 62304

C.2 Relation avec l'ISO 13485

La présente norme exige que le FABRICANT utilise un système de management de la qualité. Lorsqu'un FABRICANT utilise l'ISO 13485 [7], les exigences de l'ISO 62304 font directement référence à certaines des exigences de l'ISO 13485 comme illustré dans le Tableau C.1.

Tableau C.1 – Relation avec l'ISO 13485:2003

Article de la CEI 62304	Paragraphe correspondant de l'ISO 13485:2003
5.1 Planification du développement du logiciel	7.3.1 Planification de la conception et du développement
5.2 Analyses des exigences du logiciel	7.3.2 Eléments d'entrée de la conception et du développement
5.3 Conception architecturale du logiciel	
5.4 Conception détaillée du logiciel	
5.5 Mise en œuvre et vérification des UNITES LOGICIELLES	
5.6 Intégration et essai d'intégration du logiciel	
5.7 Essais du SYSTEME LOGICIEL	7.3.3 Résultats de sortie de la conception et du développement 7.3.4 Revue de la conception et du développement
5.8 Diffusion du logiciel	7.3.5 Vérification de la conception et du développement 7.3.6 Validation de la conception et du développement

Tableau C.1 *(suite)*

Article de la CEI 62304	Paragraphe correspondant de l'ISO 13485:2003
6.1 Etablissement du plan de maintenance du logiciel	7.3.7 Maîtrise des modifications de la conception et du développement
6.2 Analyse des problèmes et des modifications	
6.3 Mise en œuvre de la modification	7.3.5 Vérification de la conception et du développement 7.3.6 Validation de la conception et du développement
7.1 Analyse du logiciel en termes de contribution à des situations dangereuses	
7.2 MESURES DE maîtrise DU RISQUE	
7.3 VERIFICATION des MESURES DE MAITRISE DU RISQUE	
7.4 GESTION DES RISQUES des modifications du logiciel	
8.1 Identification de la configuration	7.5.3 Identification et TRAÇABILITE
8.2 Maîtrise des modifications	7.5.3 Identification et TRAÇABILITE
8.3 Documentation relative à l'état de la configuration	
9 PROCESSUS de résolution de problème logiciel	

C.3 Relation avec l'ISO 14971

Le Tableau C.2 indique les articles dans lesquels la CEI 62304 détaille les exigences du PROCESSUS de GESTION DES RISQUES exigé par l'ISO 14971.

Tableau C.2 – Relation avec l'ISO 14971:2000

Article de l'IS0 14971:2000	Article correspondant de la CEI 62304
4.1 Procédure d'ANALYSE DU RISQUE	
4.2 Usage/objet prévu et identification des caractéristiques relatives à la SECURITE du DISPOSITIF MEDICAL	
4.3 Identification des PHENOMENES DANGEREUX connus ou prévisibles	7.1 Analyse du logiciel en termes de contribution à des situations dangereuses
4.4 Estimation du (des) RISQUE(S) pour chaque phénomène dangereux	4.3 Classification de sécurité du logiciel
5 Évaluation du RISQUE	
6.1 Réduction du RISQUE	
6.2 Analyse des options	7.2.1 Définition DES MESURES DE MAITRISE DU RISQUE
6.3 Mise en œuvre des mesures de MAITRISE DU RISQUE	7.2.2 Mesures de MAITRISE DU RISQUE mises en œuvre dans le logiciel 7.3.1 Vérification des mesures de MAITRISE DU RISQUE
6.4 Évaluation du RISQUE résiduel	
6.5 ANALYSE RISQUE/bénéfice	
6.6 Autres PHENOMENES DANGEREUX engendrés	7.3.2 Consignation de toutes nouvelles séquences d'événements
6.7 Complétude de l'évaluation du RISQUE	
7 Évaluation du RISQUE résiduel global	
8 RAPPORT DE GESTION DES RISQUES	7.3.3 Consignation de la TRAÇABILITE
9 Information post-production	7.4 GESTION DES RISQUES des modifications du logiciel

C.4 Relation avec les exigences de SEMP de la CEI 60601-1:2005

C.4.1 Généralités

Les exigences applicables au logiciel sont un sous-ensemble des exigences applicables à un système électromédical programmable (SEMP). La présente norme identifie les exigences de logiciel qui viennent en supplément aux exigences de la CEI 60601-1 [1] pour les SEMP mais qui ne sont pas incompatibles avec ces exigences. Etant donné que les systèmes SEMP incluent des éléments qui ne sont pas logiciels, toutes les exigences de la CEI 60601-1 pour les SEMP ne sont pas traitées dans la présente norme.

C.4.2 Relation du logiciel avec le développement du système SEMP

En utilisant le modèle en V illustré à la Figure C.2 pour décrire les différents événements du développement d'un SEMP, on peut voir que les exigences de la présente norme de logiciel s'appliquent au niveau composant de SEMP, depuis la spécification des exigences du logiciel jusqu'à l'intégration des ELEMENTS LOGICIELS dans un SYSTEME LOGICIEL. Le SYSTEME LOGICIEL fait partie d'un sous-système électrique programmable (SSEP), qui est une partie d'un SEMP.

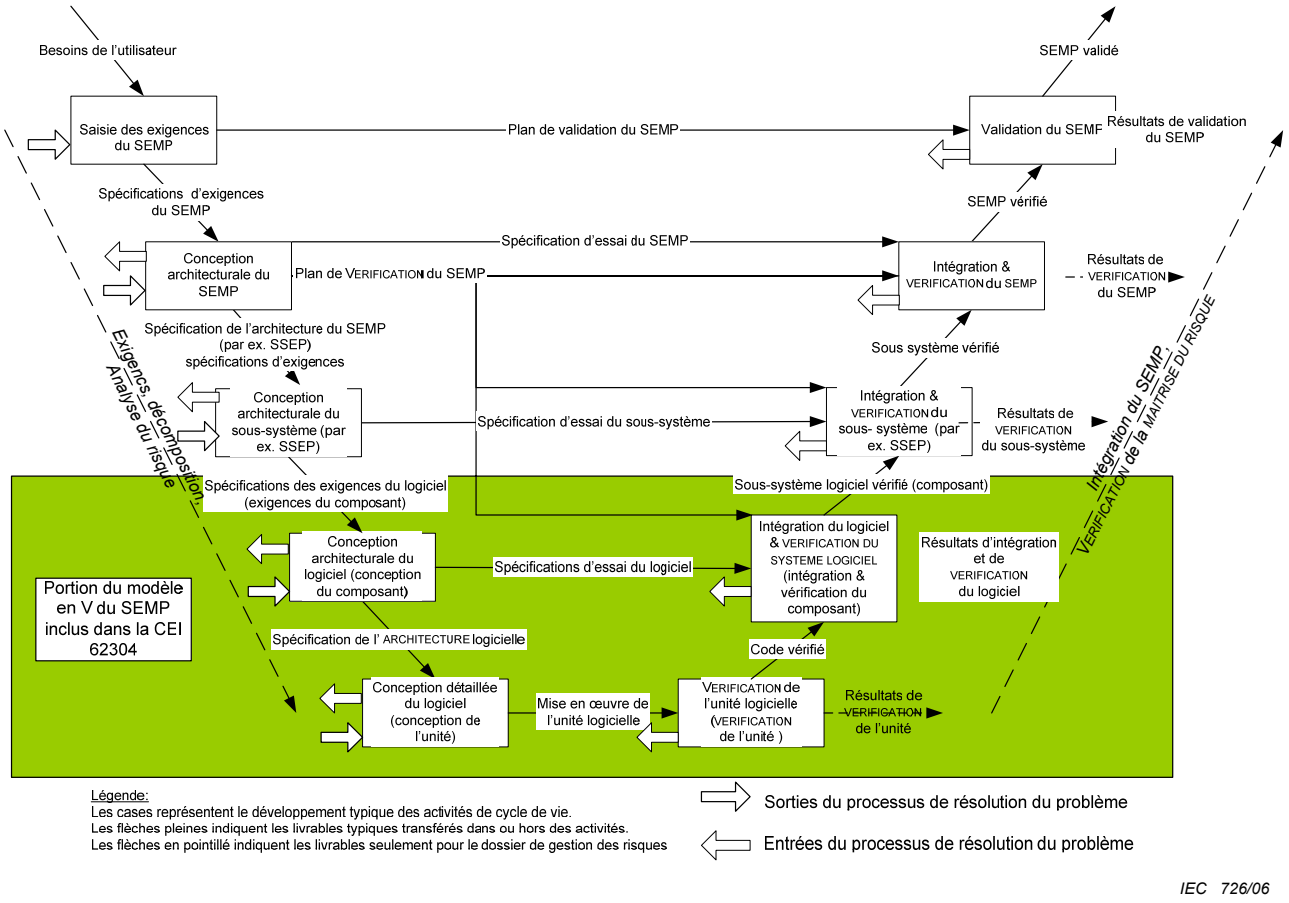

Figure C.2 – Logiciel comme partie du modèle en V

C.4.3 PROCESSUS de développement

La conformité au PROCESSUS de développement du logiciel de la présente norme (Article 5) exige qu'un plan de développement du logiciel soit spécifié et suivi; elle n'exige pas l'utilisation d'un modèle particulier de cycle de vie mais elle exige que le plan comprenne certaines ACTIVITEs et ait certains attributs. Ces exigences sont liées aux exigences du SEMP de la CEI 60601-1 pour le cycle de vie de développement, la spécification des exigences, L'ARCHITECTURE, la conception et la mise en œuvre et la VERIFICATION. Les exigences de la présente norme donnent à propos du développement du logiciel des informations plus détaillées que celles contenues dans la CEI 60601-1.

C.4.4 PROCESSUS de maintenance

La conformité au PROCESSUS de maintenance du logiciel de la présente norme (Article 6) exige que des procédures soient établies et suivies lorsque des modifications sont apportées au logiciel. Ces exigences correspondent à l'exigence de modification d'un SEMP de la CEI 60601-1. Les exigences de la présente norme fournissent, en ce qui concerne ce qui doit être effectué pour la maintenance du logiciel, des informations plus détaillées que celles des exigences pour la modification du SEMP dans la CEI 60601-1.

C.4.5 Autres PROCESSUS

Les autres PROCESSUS de la présente norme spécifient des exigences supplémentaires pour le logiciel au-delà des exigences similaires pour le SEMP dans la CEI 60601-1. Dans la plupart des cas, la CEI 60601-1 fournit une exigence d'ordre général pour le SEMP, tandis que la présente norme approfondit les PROCESSUS.

Le PROCESSUS de GESTION DES RISQUES du logiciel dans la présente norme correspond aux exigences supplémentaires de GESTION DES RISQUES identifiées pour le SEMP dans la CEI 60601-1.

Le PROCESSUS de résolution des problèmes de logiciel dans la présente norme correspond aux exigences de résolution des problèmes pour le SEMP dans la CEI 60601-1.

Le PROCESSUS de gestion de la configuration du logiciel dans la présente norme spécifie des exigences supplémentaires qui n'existent pas pour le SEMP dans la CEI 60601-1 sauf pour ce qui concerne la documentation.

C.4.6 Traitement des exigences du SEMP dans la CEI 60601-1

Le Tableau C.3 illustre les exigences SEMP de la CEI 60601-1 et les exigences correspondantes de la présente norme.

Tableau C.3 – Relation avec la CEI 60601-1

Exigences pour un SEMP de la CEI 60601-1:2005	Exigences de la CEI 62304 relatives au sous-système logiciel d'un SEMP
14.1 Généralités Les exigences de ce paragraphe doivent s'appliquer au SEMP sauf si: – le SSEP ne garantit pas la SECURITE DE BASE ni les PERFORMANCES ESSENTIELLES;ou – l'application de l'ISO 14971 démontre que la défaillance du SSEP n'entraîne pas un RISQUE inacceptable.	**4.3** Classification de sécurité du logiciel Les exigences de la CEI 60601-1 pour le système SEMP s'appliqueraient uniquement aux classes de SECURITE du logiciel B et C. La présente norme inclut certaines exigences pour la classe de SECURITE de logiciel A.
14.2 Documentation En supplément aux enregistrements et documents exigés par l'ISO 14971, les documents produits par application de l'Article 14 doivent être maintenus et faire partie DU DOSSIER DE GESTION DES RISQUES.	**4.2 GESTION DES** RISQUES
Les documents exigés par l'Article 14 doivent être revus, approuvés, édités et modifiés conformément à une procédure formelle de maîtrise des documents.	**5.1 Planification du développement du logiciel** Outre les exigences spécifiques à l'ACTIVITE de planification du développement du logiciel, il est exigé par l'ISO que soient conservés des documents qui font partie du DOSSIER DE GESTION DES RISQUES. En outre, pour les documents qui sont exigés par le système qualité, l'ISO 13485 [7]exige la maîtrise des documents.
14.3 PLAN DE GESTION DES RISQUES Le plan de gestion des RISQUES exigé au 3.5 de l'ISO 14971, doit également inclure une référence au plan de validation du SEMP (voir 14.11).	N'est pas spécifiquement exigé. Il n'y a pas de plan spécifique de validation du logiciel. Le plan de validation du SEMP se situe au niveau du système et ainsi il est hors du domaine d'application de la présente norme de logiciel. La présente norme exige une TRAÇABILITE depuis le PHENOMENE DANGEREUX en passant par la cause spécifique au logiciel et les mesures de MAITRISE DU RISQUE jusqu'à la VERIFICATION de la mesure de MAITRISE DU RISQUE (voir 7.3)
14.4 Cycle de développement du système SEMP Le cycle de développement d'un SEMP doit être documenté.	**5.1 Planification du développement du logiciel** 5.1.1 Plan de développement du logiciel Les éléments couverts par le plan de développement du logiciel constituent un cycle de développement du logiciel.
Le cycle de développement d'un SEMP doit comporter un ensemble de jalons bien défini.	
A chaque jalon, les ACTIVITES qui doivent être menées à bien et les méthodes de vérification à appliquer à ces ACTIVITES, doivent être définies.	5.1.6 Planning de VERIFICATION du logiciel Les TACHES DE VERIFICATION, les critères de réception et d'étapes doivent être planifiés
Chaque ACTIVITE doit être définie en indiquant ses éléments d'entrée et de sortie.	5.1.1 Plan de développement du logiciel Les ACTIVITES sont définies dans la présente norme. La documentation à produire est définie dans chaque ACTIVITE.
Chaque jalon doit identifier les ACTIVITEs de gestion des RISQUES qui doivent être menées à bien avant ce jalon.	
Le cycle de développement d'un SEMP doit être adapté à chaque développement spécifique en élaborant des plans qui détaillent les ACTIVITES, des jalons et des plannings.	5.1.1 Plan de développement du logiciel La présente norme permet de documenter le cycle de vie de développement dans le plan de développement. Ce qui signifie que le plan de développement contient un cycle de vie de développement adapté.
Le cycle de développement d'un SEMP doit inclure les exigences en matière de documentation.	5.1.1 Plan de développement du logiciel 5.1.8 Planification de la documentation
14.5 Résolution des problèmes Le cas échéant, un système documenté de résolution des problèmes pendant et entre toutes les phases et ACTIVITES du cycle de développement d'un système SEMP doit être développé et maintenu.	**9 Processus de résolution de** problème logiciel

Tableau C.3 *(suite)*

Exigences pour un SEMP de la CEI 60601-1:2005	**Exigences de la CEI 62304 relatives au sous-système logiciel d'un SEMP**
En fonction du type de produit, le système de résolution des problèmes peut: – être consigné par écrit en tant que partie du cycle de développement du SEMP; – permettre de rendre compte de problèmes potentiels ou existants affectant la SECURITE DE BASE OU les PERFORMANCES ESSENTIELLES – inclure une évaluation de chaque problème pour les RISQUES associés; – identifier les critères à satisfaire pour prononcer une conclusion; – identifier les actions à entreprendre pour résoudre chaque problème.	5.1.1 Plan de développement du logiciel 9.1 Elaboration des RAPPORTS DE PROBLEME
14.6 PROCESSUS de GESTION DES RISQUES	**7 Processus DE GESTION DES** RISQUES DU LOGICIEL
14.6.1 Identification des PHENOMENES DANGEREUX connus ou prévisibles Lors de l'élaboration de la liste des PHENOMENES DANGEREUX connus ou prévisibles, le FABRICANT doit tenir compte des PHENOMENES DANGEREUX liés aux aspects logiciels et matériels du système SEMP y compris ceux relatifs aux liaisons réseau/données, composants en provenance de tierce partie et sous-systèmes hérités.	7.1 Analyse du logiciel en termes de contribution à des situations dangereuses La présente norme ne mentionne pas spécifiquement les liaisons réseau/données
14.6.2 MAITRISE DU RISQUE Des outils et des procédures correctement validés doivent être sélectionnés et identifiés pour la mise en œuvre de chaque mesure de MAITRISE DU RISQUE. Ces outils et procédures doivent convenir pour s'assurer que chaque mesure de MAITRISE DU RISQUE réduit de manière satisfaisante le(s) risque(s) identifié(s).	5.1.4 Planification des normes, méthodes et outils de développement du logiciel La présente norme exige l'identification d'outils et de méthodes spécifiques à utiliser pour le développement de manière générale et non pour chaque mesure de MAITRISE DU RISQUE.
14.7 Spécification des exigences Pour le SEMP et chacun de ses sous-systèmes (par exemple pour un sous-système SSEP), il doit exister une spécification des exigences.	**5.2 Analyses des exigences** du logiciel La présente norme traite uniquement des sous-SYSTEMES LOGICIELS d'un SEMP.
La spécification des exigences pour un SYSTEME ou un sous-système doit inclure et faire la distinction entre une éventuelle performance essentielle et les mesures de MAITRISE DU RISQUE mises en œuvre par ledit système ou sous-système.	5.2.1 Définition et documentation des exigences du logiciel d'après les exigences du SYSTEME 5.2.2 Teneur des exigences du logiciel 5.2.3 Intégration des mesures de MAITRISE DU RISQUE dans les exigences du logiciel La présente norme n'exige pas que les exigences liées à des performances essentielles et à des mesures de MAITRISE DU RISQUE soient distinguées des autres exigences mais elle exige en effet que toutes les exigences soient identifiées de manière univoque.

Tableau C.3 *(suite)*

Exigences pour un SEMP de la CEI 60601-1:2005	Exigences de la CEI 62304 relatives au sous-système logiciel d'un SEMP
14.8 Architecture Pour le SEMP et chacun de ses sous-systèmes, une ARCHITECTURE doit être spécifiée et satisfaire à la spécification des exigences.	**5.3** Conception ARCHITECTURALE du logiciel
Le cas échéant, pour réduire le RISQUE à un niveau acceptable, la spécification de l'ARCHITECTURE doit utiliser: a) des composants ayant des caractéristiques de haute intégrité; b) des fonctions à sécurité positive; c) la redondance; d) la diversité; e) le découpage par fonctionnalité; f) une conception défensive, par exemple les limites des effets dangereux potentiels en réduisant la puissance de sortie disponible ou en introduisant des moyens pour limiter le déplacement des organes de manœuvre.	5.3.5 Identification des séparations nécessaires à la MAITRISE DU RISQUE Le découpage est la seule technique identifiée et cela uniquement parce qu'il est exigé d'indiquer la manière dont l'intégrité du découpage est assurée.
La spécification de l'ARCHITECTURE doit tenir compte: g) de l'allocation des mesures de MAITRISE DES RISQUEs aux sous-systèmes et éléments du SEMP; h) des types de pannes des éléments et leurs conséquences; i) des pannes ayant les mêmes causes; j) des pannes systématiques; k) de la durée des intervalles entre les essais et la couverture du diagnostic; l) de la maintenabilité; m) de la protection contre des usages abusifs raisonnablement prévisibles; n) de la spécification des liaisons réseau/données le cas échéant.	Ceci n'est pas inclus dans la présente norme.
14.9 Conception et mise en œuvre Le cas échéant, la conception doit être scindée en sous-systèmes, chaque sous-système ayant une spécification pour la conception et pour les essais.	**5.4 Conception détaillée du logiciel** 5.4.2 Elaboration de la conception détaillée de chaque UNITE LOGICIELLE La présente norme n'exige pas une spécification d'essai pour la conception détaillée.
Les données descriptives concernant l'environnement de la conception doivent être comprises dans le DOSSIER DE GESTION DES RISQUES.	5.4.2 Elaboration de la conception détaillée de chaque UNITE LOGICIELLE
14.10 VERIFICATION La VERIFICATION est exigée pour toutes les fonctions mettant en œuvre la SECURITE DE BASE, les PERFORMANCES ESSENTIELLES ou des mesures de MAITRISE DU RISQUE.	**5.1.6** Planification de la VERIFICATION du logiciel La VERIFICATION est requise pour chaque ACTIVITE.

Tableau C.3 *(suite)*

Exigences pour un SEMP de la CEI 60601-1:2005	Exigences de la CEI 62304 relatives au sous-système logiciel d'un SEMP
Un plan de VERIFICATION doit être établi pour indiquer comment ces fonctions doivent être vérifiées. Le plan doit comprendre: – Les jalons d'exécution des vérifications pour chaque fonction; – Le choix et la documentation des stratégies, ACTIVITEs, techniques de vérification ainsi que le niveau approprié d'indépendance du personnel chargé des vérifications; – Le choix et l'utilisation des outils de vérification; – Les critères de couverture de la VERIFICATION.	5.1.6 Planification de la VERIFICATION du logiciel L'indépendance du personnel n'est pas incluse dans la présente norme. Elle est considérée prise en charge dans l'ISO 13485.
La VERIFICATION doit être conduite conformément au plan de VERIFICATION. Les résultats des ACTIVITEs de VERIFICATION doivent être documentés.	Les exigences de VERIFICATION concernent la plupart des ACTIVITEs.
14.11 Validation du SEMP Un plan de validation du SEMP doit inclure la validation de la SECURITE fondamentale et des performances essentielles et doit exiger des contrôles de tout fonctionnement imprévu du SEMP.	La présente norme ne couvre pas la validation du logiciel. La VALIDATION du SEMP est une ACTIVITE au niveau SYSTEME et ne s'inscrit pas dans le domaine d'application de la présente norme.
La VALIDATION du SEMP doit être effectuée conformément au plan de VALIDATION du SEMP. Les résultats des ACTIVITEs de VALIDATION du SEMP doivent être documentés.	La présente norme ne couvre pas la validation du logiciel. La VALIDATION du SEMP est une ACTIVITE au niveau SYSTEME et ne s'inscrit pas dans le domaine d'application de la présente norme.
La personne entièrement responsable de la validation du SEMP doit être indépendante de l'équipe de conception. Le FABRICANT doit justifier par écrit le niveau d'indépendance.	La présente norme ne couvre pas la validation du logiciel. La VALIDATION du SEMP est une ACTIVITE au niveau SYSTEME et ne s'inscrit pas dans le domaine d'application de la présente norme.
La validation d'un SEMP ne doit pas être confiée à un membre d'une équipe de conception ayant réalisé la conception du système SEMP.	La présente norme ne couvre pas la validation du logiciel. La VALIDATION du SEMP est une ACTIVITE au niveau SYSTEME et ne s'inscrit pas dans le domaine d'application de la présente norme.
Toutes les relations professionnelles entre les membres de l'équipe de validation de SEMP et les membres de l'équipe de conception doivent figurer dans le DOSSIER DE GESTION DES RISQUES.	La présente norme ne couvre pas la validation du logiciel. La VALIDATION du SEMP est une ACTIVITE au niveau SYSTEME et ne s'inscrit pas dans le domaine d'application de la présente norme.
Le DOSSIER DE GESTION DES RISQUES doit comporter une référence aux méthodes et résultats de la validation du SEMP.	La présente norme ne couvre pas la validation du logiciel. La VALIDATION du SEMP est une ACTIVITE au niveau SYSTEME et ne s'inscrit pas dans le domaine d'application de la présente norme.
14.12 Modification Si tout ou partie d'une conception résulte de la modification d'une conception antérieure, l'ensemble des dispositions du présent paragraphe s'applique comme s'il s'agissait d'une nouvelle conception ou, la validité conservée de toute la documentation de conception préalable doit être évaluée selon une procédure documentée de modification.	**6 Processus de maintenance du logiciel** La présente norme considère par principe qu'il convient de planifier la maintenance du logiciel et qu'il est recommandé que la mise en œuvre des modifications utilise le PROCESSUS de développement du logiciel ou un PROCESSUS établi de maintenance du logiciel.

Tableau C.3 *(suite)*

Exigences pour un SEMP de la CEI 60601-1:2005	Exigences de la CEI 62304 relatives au sous-système logiciel d'un SEMP
14.13 Connexion entre le SEMP et d'autres équipements par liaisons réseau/données S'il est prévu de connecter le SEMP à un autre équipement par une liaison réseau/données qui n'est pas maîtrisée par le FABRICANT du SEMP, LA description technique DOIT: a) spécifier les caractéristiques de la liaison réseau/données nécessaire pour que le SEMP puisse satisfaire à son usage/objet prévu; b) énumérer les PHENOMENES DANGEREUX potentiels résultant d'une défaillance de la liaison réseau/données à fournir les caractéristiques spécifiées; c) informer l'organisation responsable que: – la connexion du SEMP à une liaison réseau/données qui comprend d'autres équipements pourrait entraîner des RISQUES précédemment non identifiés pour les patients, opérateurs ou tierces parties; – il convient que l'organisation responsable identifie, analyse, évalue et maîtrise ces RISQUES; – les modifications ultérieures à la liaison réseau/données pourrait introduire de nouveaux RISQUES et nécessiter une analyse supplémentaire; et – les modifications de la liaison réseau/données comprennent: ▪ les modifications de la configuration de la liaison réseau/données ▪ la connexion d'éléments supplémentaires à la liaison réseau/données ▪ la déconnexion d'éléments de la liaison réseau/données ▪ la mise à jour d'équipements connectés à la liaison réseau/données ▪ la mise à niveau d'équipements connectés à la liaison réseau/données	Les exigences relatives à la liaison réseau/données ne sont pas incluses dans la présente norme.

C.4.7 Relation avec les exigences de la CEI 60601-1-4

La CEI 60601-1-4 continuera à être utilisée jusqu'à ce que la période transitoire de la CEI 60601-1:2005 soit achevée.

Le Tableau C.4 illustre les exigences de la CEI 60601-1-4 [2] et les exigences correspondantes de la présente norme. Ceci ne signifie pas que les exigences correspondantes de la présente norme couvrent pleinement les exigences de la CEI 60601-1-4. De nombreuses parties des exigences de la norme 60601-1-4 sont couvertes par conformité à l'ISO 14971. Certaines exigences de la CEI 60601-1-4 ne sont pas traitées par la CEI 62304.

Tableau C.4 – Relation avec la CEI 60601-1-4

Exigences POUR LE SYSTEME SEMP de la CEI 60601-1-4:1996 plus l'amendement 1:1999	Exigences correspondantes de la CEI 62304
6.8 Documents d'accompagnement	
6.8.201	4.2 et 4.3 c)
52.201 Documentation	
52.201.1	4.1
52.201.2	4.1 et 4.2

Tableau C.4 *(suite)*

Exigences pour le système SEMP de la CEI 60601-1-4:1996 plus l'amendement 1:1999	Exigences correspondantes de la CEI 62304
52.201.3	4.2
52.202 PLAN DE GESTION DES RISQUES	
52.202.1	4.2
52.202.2	5.1.1,5.1.5
52.202.3	4.1, 5.1.2
52.203 Cycle de développement	
52.203.1	5.1.1
52.203.2	5.1.1
52.203.3	
52.203.4	5.1.7
52.203.5	7
52.204 Traitement de la gestion des risques	
52.204.1	4.2
52.204.2	4.2, 7
52.204.3	
52.204.3.1	
52.204.3.1.1	4.2,7.1
52.204.3.1.2	4.2,7.1.2
52.204.3.1.3	4.2
52.204.3.1.4	4.2, 7.1.2 e)
52.204.3.1.5	4.2, 7.1.2
52.204.3.1.6	4.2,7.1
52.204.3.1.7	4.2
52.204.3.1.8	4.2
52.204.3.1.9	4.2
52.204.3.1.10	4.2
52.204.3.2	
52.204.3.2.1	4.2
52.204.3.2.2	4.2,4.3
52.204.3.2.3	
52.204.3.2.4	
52.204.3.2.5	4.2
52.204.4	
52.204.4.1	4.2
52.204.4.2	4.2
52.204.4.3	4.2
52.204.4.4	4.2
52.204.4.5	
52.204.4.6	4.2

Tableau C.4 *(suite)*

Exigences pour le système SEMP de la CEI 60601-1-4:1996 plus l'amendement 1:1999	Exigences correspondantes de la CEI 62304
52.205 Qualification du personnel	4.1
52.206 Spécification des exigences	
52.206.1	5.2
52.206.2	7.2.2
52.206.3	
52.207 Architecture	
52.207.1	5.3.1
52.207.2	5.3
52.207.3	
52.207.4	
52.207.5	
52.208 Conception et réalisation	
52.208.1	5
52.208.2	
52.209 Vérification	
52.209.1	5.7.1
52.209.2	5.1.5, 5.1.6
52.209.3	5.2.6, 5.3. 6, 5.4.4, 5.5.5, 5.6, 5.7
52.209.4	
52.210 Validation	
52.210.1	4.1
52.210.2	4.1
52.210.3	4.1
52.210.4	
52.210.5	
52.210.6	
52.210.7	
52.211 Modification	
52.211.1	6
52.211.2	4.1,6
52.212 Évaluation	
52.212.1	4.1

C.5 Relation avec la CEI 61010-1

Le domaine d'application de la CEI 61010-1 [4] couvre les appareils électriques d'essai, de mesurage, de régulation et de laboratoire. Seule une partie des appareils de laboratoire est utilisée dans un environnement vertical ou comme matériel de diagnostic *in vitro* (IVD).

Du fait des réglementations légales ou des références normatives, le matériel IVD est assimilé aux DISPOSITIFS MEDICAUX sans cependant s'inscrire dans le domaine d'application de la CEI 60601-1 [1]. Ceci est imputable non seulement au fait que strictement parlant, les instruments IVD ne sont pas des DISPOSITIFS MEDICAUX au contact direct avec les patients mais également au fait que ces produits sont fabriqués pour de nombreuses applications différentes dans divers laboratoires. L'utilisation en tant qu'appareil IVD ou en tant qu'accessoire pour appareil IVD est donc rare.

Si le matériel de laboratoire est utilisé en tant qu'appareil IVD, les résultats mesurés obtenus doivent être évalués conformément à des critères médicaux. L'application de l'ISO 14971 est exigée pour la GESTION DES RISQUES. Si ces produits comportent également des logiciels qui peuvent donner lieu à des PHENOMENES DANGEREUX, lorsque par exemple, une défaillance due au logiciel peut entraîner une modification indésirable des données médicales (résultats des mesures), la CEI 62304 doit être prise en compte.

L'organigramme de la Figure C.3 constitue une aide utile pour expliquer le principe du PROCESSUS de GESTION DES RISQUES et l'application de la CEI 62304.

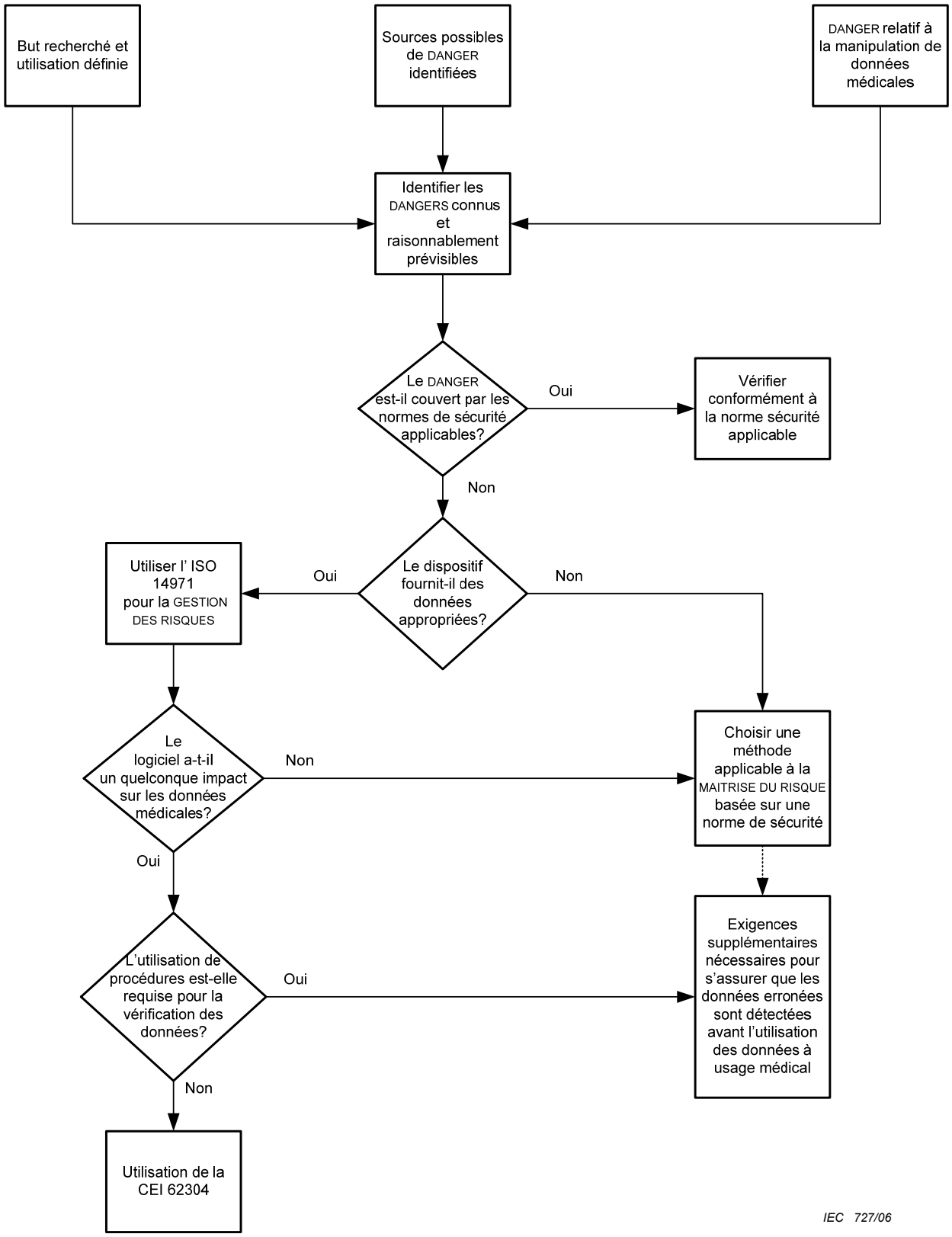

Figure C.3 – Application de la CEI 62304 avec la CEI 61010-1

C.6 Relation avec l'ISO/CEI 12207

La présente norme est issue de l'approche et des concepts de l'ISO/CEI 12207 [9], qui définit les exigences applicables au PROCESSUS de cycle de vie des logiciels en général, c'est-à-dire sans restriction aux DISPOSITIFS MEDICAUX.

Les principales différences de la présente norme par rapport à l'ISO/CEI 12207 sont les suivantes. Elle:

- exclue des aspects SYSTEME tels que les exigences système, l'ARCHITECTURE et la validation système;
- omet certains PROCESSUS tels que la duplication des ACTIVITES qui est documentée ailleurs pour les DISPOSITIFS MEDICAUX;
- ajoute le PROCESSUS de GESTION DES RISQUES (SECURITE) et le PROCESSUS de diffusion des logiciels;
- incorpore la documentation et la vérification qui viennent à l'appui des PROCESSUS de développement et de maintenance;
- fusionne les ACTIVITEs de mise en œuvre et de planification de chaque PROCESSUS en une seule ACTIVITE dans les PROCESSUS de développement et de maintenance;
- classe les exigences par rapport aux besoins en matière de SECURITE; et
- ne classe pas explicitement les PROCESSUS comme principaux ou secondaires, ni ne les groupe comme le fait l'ISO/CEI 12207.

La plupart de ces modifications résultent du souhait d'adapter la norme aux besoins du secteur des dispositifs médicaux:

- en se concentrant sur les aspects SECURITE et la GESTION DES RISQUES du DISPOSITIF MEDICAL de l'ISO 14971;
- en sélectionnant les PROCESSUS qui conviennent, lorsqu'ils sont utiles dans un environnement réglementé;
- en tenant compte du fait que le développement du logiciel s'intègre dans un système qualité (qui couvre certains des PROCESSUS et exigences de l'ISO/CEI 12207); et
- en réduisant le niveau d'abstraction pour en faciliter l'utilisation.

La présente norme ne comporte pas de contradiction avec l'ISO/CEI 12207. Cette dernière peut être utile comme aide à l'établissement d'un modèle de cycle de vie de développement du logiciel bien structuré qui incorpore les exigences de la présente norme.

Le Tableau C.5, qui a été préparé par le JTC1/SC7 de l'ISO/CEI, illustre la relation entre la CEI 62304 et l'ISO/CEI 12207.

Tableau C.5 – Relation avec l'ISO/CEI 12207

Processus de l'ISO/CEI 62304		Processus de l'ISO/CEI 12207	
Activité	**Tâche**	**Activité**	**Tâche**
5 PROCESSUS de développement du logiciel		5.3 Processus de développement 6.1 Processus de documentation 6.2 Processus de la gestion de la configuration 6.4 Processus de la vérification 6.5 Processus de la validation 6.8 Processus de résolution de problème 7.1 Processus de la gestion	
5.1 Planification du développement du logiciel		5.3.1 Mise en œuvre du processus 5.3.3 Conception architecturale du système 5.3.7 Codage et test du logiciel 5.3.8 Intégration du logiciel 5.3.9 Test de qualification du logiciel 5.3.10 Intégration du système 6.1.1 Mise en œuvre du processus 6.2.1 Mise en œuvre du processus 6.2.2 Identification de la configuration 6.4.1 Mise en œuvre du processus 6.5.1 Mise en œuvre du processus 6.8.1 Mise en œuvre du processus 7.1.2 Planification 7.1.3 Exécution et commande 7.2.2 Etablissement de l'infrastructure 7.2.3 Maintenance de l'infrastructure	
	5.1.1 Plan de développement du logiciel	5.3.1 Mise en œuvre du processus 7.1.2 Planification	5.3.1.1 5.3.1.3 5.3.1.4 7.1.2.1
	5.1.2 Mise à jour du plan de développement logiciel	7.1.3 Exécution et commande.	7.1.3.3
	5.1.3 Référence du plan de développement du logiciel à la conception et au développement du SYSTEME	5.3.3 Conception architecturale du système 5.3.10 Intégration du système. 6.5.1 Mise en œuvre du processus	5.3.3.1 5.3.10.1 6.5.1.4
	5.1.4 Planification des normes, méthodes et outils de développement du logiciel	5.3.1 Mise en œuvre du processus	5.3.1.3 5.3.1.4
	5.1.5 Planification de l'intégration du logiciel et des essais d'intégration	5.3.8 Intégration du logiciel.	5.3.8.1
	5.1.6 Planification de la VERIFICATION du logiciel	6.4.1 Mise en œuvre du processus 5.3.7 Codage et test du logiciel 5.3.8 Intégration du logiciel 5.3.9 Test de qualification du logiciel	6.4.1.4 6.4.1.5 5.3.7.5 5.3.8.5 5.3.9.3
	5.1.7 Planification de la GESTION DES RISQUES du logiciel	Amd.1:2002 – F 3.1.5 Processus de la gestion des risques	
	5.1.8 Planification de la documentation	6.1.1 Mise en œuvre du processus	6.1.1.1
	5.1.9 Planification de la gestion de configuration du logiciel	6.2.1 Mise en œuvre du processus 6.8.1 Mise en œuvre du processus	6.2.1.1 6.8.1.1
	5.1.10 Eléments annexes à contrôler	7.2.2 Etablissement de l'infrastructure 7.2.3 Maintenance de l'infrastructure	7.2.2.1 7.2.3.1
	5.1.11 Eléments de contrôle de la configuration du logiciel avant VERIFICATION	6.2.2 Identification de la configuration	6.2.2.1

Tableau C.5 *(suite)*

Processus de l'ISO/CEI 62304		Processus de l'ISO/CEI 12207	
Activité	**Tâche**	**Activité**	**Tâche**
5.2 Analyses des exigences du logiciel		5.3.3 Conception architecturale du système 5.3.4 Analyse des exigences du logiciel 6.4.2 Vérification	
	5.2.1 Définition et documentation des exigences du logiciel d'après les exigences du SYSTEME	5.3.3 Conception architecturale du système	5.3.3.1
	5.2.2 Teneur des exigences du logiciel	5.3.4 Analyse des exigences du logiciel	5.3.4.1
	5.2.3 Intégration des mesures de MAITRISE DU RISQUE dans les exigences du logiciel		
	5.2.4 R Réévaluation de l'ANALYSE DU RISQUE du DISPOSITIF MEDICAL e		Aucune
	5.2.5 Mise à jour des exigences du SYSTEME	5.3.4 Analyse des exigences du logiciel	a) b)
	5.2.6 Vérification des exigences du logiciel	5.3.4 Analyse des exigences du logiciel 6.4.2 Vérification	5.3.4.2 6.4.2.3
5.3 Conception ARCHITECTURALE du logiciel		5.3.5 Conception architecturale du logiciel	
	5.3.1 Conversion des exigences du logiciel en ARCHITECTURE	5.3.5 Conception architecturale du logiciel	5.3.5.1
	5.3.2 Elaboration d'une ARCHITECTURE pour les interfaces d'ELEMENTS LOGICIELS		5.3.5.2
	5.3.3 Spécification des exigences fonctionnelles et de performance des éléments logiciels SOUP		Aucune
	5.3.4 Spécification des matériels et des logiciels SYSTEME nécessaires à l'élément logiciel SOUP		Aucune
	5.3.5 Identification des séparations nécessaires à la MAITRISE DU RISQUE		Aucune
	5.3.6 Vérification de L'ARCHITECTURE du logiciel	5.3.5 Conception architecturale du logiciel	5.3.5.6
5.4 Conception détaillée du logiciel		5.3.6 Conception détaillée du logiciel 6.4.2 Vérification	
	5.4.1 Décomposition de l'ARCHITECTURE des LOGICIELS en UNITES LOGICIELLES	5.3.6 Conception détaillée du logiciel	5.3.6.1
	5.4.2 Elaboration de la conception détaillée de chaque UNITE LOGICIELLE		
	5.4.3 Elaboration de la conception détaillée pour les interfaces		5.3.6.2
	5.4.4 Vérification de la conception détaillée	6.4.2 Vérification	5.3.6.7
5.5 Mise en œuvre et vérification des UNITES LOGICIELLES		5.3.6 Conception détaillée du logiciel 5.3.7 Codage et test du logiciel 6.4.2 Vérification	
	5.5.1 Mise en œuvre de chaque UNITE LOGICIELLE	5.3.7 Codage et test du logiciel	5.3.7.1
	5.5.2 Etablissement du PROCESSUS DE VERIFICATION DES UNITES LOGICIELLES	5.3.6 Conception détaillée du logiciel 5.3.7 Codage et test du logiciel	5.3.6.5 5.3.7.5
	5.5.3 Critères d'acceptation de l'UNITE LOGICIELLE	5.3.7 Codage et test du logiciel	5.3.7.5
	5.5.4 Critères supplémentaires d'acceptation de l'UNITE LOGICIELLE	5.3.7 Codage et test du logiciel 6.4.2 Vérification	5.3.7.5 6.4.2.5

Tableau C.5 *(suite)*

Processus de l'ISO/CEI 62304		Processus de l'ISO/CEI 12207	
Activité	**Tâche**	**Activité**	**Tâche**
	5.5.5 VERIFICATION de l'UNITE LOGICIELLE	5.3.7 Codage et test du logiciel	5.3.7.2
5.6 Intégration et essai d'intégration du logiciel		5.3.8 Intégration du logiciel 5.3.9 Essai de qualification du logiciel 5.3.10 Intégration du système 6.4.1 Mise en œuvre du processus 6.4.2 Vérification	
	5.6.1 Intégration des UNITES LOGICIELLES	5.3.8 Intégration du logiciel	5.3.8.2
	5.6.2 Vérification de l'intégration du logiciel	5.3.8 Intégration du logiciel 5.3.10 Intégration du système	5.3.8.2 5.3.10.1
	5.6.3 Essai du logiciel intégré	5.3.9 Essai de qualification du logiciel.	5.3.9.1
	5.6.4 Teneur des essais d'intégration		5.3.9.3
	5.6.5 Vérification des procédures d'essais d'intégration	6.4.2 Vérification	6.4.2.2
	5.6.6 Réalisation d'essais de régression	5.3.8 Intégration du logiciel	5.3.8.2
	5.6.7 Teneur de l'enregistrement des essais d'intégration	5.3.8 Intégration du logiciel	5.3.8.2
	5.6.8 Utilisation du PROCESSUS de résolution des problèmes de logiciel	6.4.1 Mise en œuvre du processus	6.4.1.6
5.7 Essais du SYSTEME LOGICIEL		5.3.8 Intégration du logiciel 5.3.9 Essai de qualification du logiciel 6.4.1 Mise en œuvre du processus 6.4.2 Vérification 6.8.1 Mise en œuvre du processus	
	5.7.1 Etablissement d'essais pour les exigences du logiciel	5.3.8 Intégration du logiciel 5.3.9 Essai de qualification du logiciel	5.3.8.4 5.3.9.1
	5.7.2 Utilisation du PROCESSUS de résolution des problèmes de logiciel	6.4.1 Mise en œuvre du processus	6.4.1.6
	5.7.3 Contre-essais après modifications	6.8.1 Mise en œuvre du processus	6.8.1.1
	5.7.4 Vérification des essais du SYSTEME LOGICIEL	6.4.2 Vérification 5.3.9 Essai de qualification du logiciel	6.4.2.2 5.3.9.3
	5.7.5 Teneur des enregistrements d'essai du SYSTEME LOGICIEL	5.3.9 Essai de qualification du logiciel	5.3.9.1
5.8 Diffusion du logiciel		5.3.9 Essai de qualification du logiciel 5.4.2 Essai opérationnel 6.2.5 Évaluation de la configuration 6.2.6 Gestion de la diffusion et livraison	
	5.8.1 Assurance de l'achèvement de la VERIFICATION du logiciel	5.4.2 Essai opérationnel 6.2.6 Gestion de la diffusion et livraison	5.4.2.1 5.4.2.2 6.2.6.1
	5.8.2 Consignation des ANOMALIES résiduelles connues	6.2.5 Évaluation de la configuration 5.3.9 Essai de qualification du logiciel	6.2.5.1 5.3.9.3
	5.8.3 Évaluation des ANOMALIES résiduelles connues		
	5.8.4 Consignation des VERSIONS diffusées	6.2.6 Gestion de la diffusion et livraison	6.2.6.1
	5.8.5 Consignation de la manière dont le logiciel diffusé a été créé		
	5.8.6 Assurance de l'achèvement complet des ACTIVITES et des TACHES		
	5.8.7 Archivage du logiciel		

Tableau C.5 *(suite)*

Processus de l'ISO/CEI 62304		Processus de l'ISO/CEI 12207	
Activité	**Tâche**	**Activité**	**Tâche**
	5.8.8 Assurance de la reproductibilité du logiciel diffusé		
6 PROCESSUS de Maintenance du logiciel		5.5 Processus de maintenance 6.2 Processus de la gestion de la configuration	
6.1 Etablissement du plan de maintenance du logiciel		5.5.1 Mise en œuvre du processus	5.5.1.1
6.2 Analyse des problèmes et des modifications		5.5.1 Mise en œuvre du processus 5.5.2 Analyse des problèmes et des modifications 5.5.3 Mise en œuvre de la modification 5.5.5 Migration	
	6.2.1 Consignation et évaluation des retours d'information		
	6.2.1.1 Contrôle des retours d'information	5.5.1 Mise en œuvre du processus	5.5.1.1 5.5.1.2
	6.2.1.2 Consignation et évaluation des retours d'information		
	6.2.1.3 Évaluation des influences des RAPPORTS DE PROBLEME sur la SECURITE	5.5.2 Analyse des problèmes et des modifications	5.5.2.1 5.5.2.2 5.5.2.3 5.5.2.4
	6.2.2 Utilisation du PROCESSUS de résolution des problèmes du logiciel	5.5.1 Mise en œuvre du processus	5.5.1.2
	6.2.3 Analyse des demandes de modification	5.5.2 Analyse des problèmes et des modifications	5.5.2.1
	6.2.4 Approbation des demandes de modification	5.5.2 Analyse des problèmes et des modifications	5.5.2.5
	6.2.5 Communication aux utilisateurs et aux organismes de réglementation	5.5.3 Mise en œuvre de la modification 5.5.5 Migration	5.5.3.1 5.5.5.3
6.3 Mise en œuvre de la modification		5.5.3 Mise en œuvre de la modification 6.2.6 Gestion de la diffusion et livraison	
	6.3.1 Utilisation d'un PROCESSUS établi pour mettre en œuvre la modification	5.5.3 Mise en œuvre de la modification	5.5.3.2
	6.3.2 Rediffusion du SYSTEME LOGICIEL modifié	6.2.6 Gestion de la diffusion et livraison	6.2.6.1
7 PROCESSUS DE GESTION DES RISQUES du logiciel		Amd.1:2002 – F 3.15 Processus de la gestion des risques Le processus dans la 62304 traite des problèmes liés au risque/danger qui ne sont pas traités dans l'amendement 1. Des caractères communs existent (mesures contre le risque, etc), mais l'analyse s'oriente d'une manière totalement différente.	
8 PROCESSUS de gestion de configuration du Logiciel		5.5 Processus de maintenance 6.2 Processus de la gestion de la configuration	
8.1 Identification de la configuration		6.2.2 Identification de la configuration	
	8.1.1 Etablissement des moyens d'identification des ELEMENTS DE CONFIGURATION	6.2.2 Identification de la configuration	6.2.2.1
	8.1.2 Identification des logiciels SOUP		Aucune
	8.1.3 Identification de la documentation de configuration du SYSTEME	6.2.2 Identification de la configuration	6.2.2.1

Tableau C.5 *(suite)*

Processus de l'ISO/CEI 62304		Processus de l'ISO/CEI 12207	
Activité	**Tâche**	**Activité**	**Tâche**
8.2 Maîtrise des modifications		5.5.3 Mise en œuvre de la modification 6.2.3 Maîtrise de la configuration	
	8.2.1 Approbation des DEMANDES DE MODIFICATION	6.2.3 Maîtrise de la configuration	6.2.3.1
	8.2.2 Mise en œuvre des modifications	5.5.3 Mise en œuvre de la modification 6.2.3 Maîtrise de la configuration	5.5.3.2 6.2.3.1
	8.2.3 Vérification des modifications	6.2.3 Maîtrise de la configuration	6.2.3.1
	8.2.4 Prévision des moyens de TRAÇABILITE de la modification		
8.3 Documentation relative à l'état de la configuration		6.2.4 Documentation relative à l'état de la configuration	6.2.4.1
9 PROCESSUS de résolution de problème logiciel		5.5 Processus de maintenance 6.2 Gestion de la configuration 6.8 Processus de résolution du problème	
9.1 Elaboration des RAPPORTS DE PROBLEME		6.8.1 Mise en œuvre du processus 6.8.2 Résolution du problème	6.8.1.1 b) 6.8.2.1
9.2 Etude du problème		6.8.2 Résolution du problème 6.8.1 Mise en œuvre du processus	6.8.2.1 6.8.1.1 b)
9.2 Information des parties concernées		6.8.1 Mise en œuvre du processus	6.8.1.1 a)
9.3 Utilisation du processus de la maîtrise des modifications		6.2.3 Maîtrise de la configuration. 5.5.3 Mise en œuvre de la modification	
9.4 Conservation des enregistrements		6.8.1 Mise en œuvre du processus	6.8.1.1 a)
9.6 Analyse de tendance pour les problèmes		6.8.1 Mise en œuvre du processus 6.8.2 Résolution du problème	6.8.1.1 b) 6.8.2.1
9.7 VERIFICATION de la résolution des problèmes du logiciel		6.8.1 Mise en œuvre du processus	6.8.1.1 d)
9.8 Teneur de la documentation d'essai			Toutes les tâches d'essais de la 12207 nécessitent une consignation

C.7 Relation avec la CEI 61508

La question a été soulevée quant à savoir s'il convient que la présente norme, qui concerne la conception de logiciels critiques pour la SECURITE, suive les principes de la CEI 61508. La position de la présente norme est expliquée ci-dessous.

La CEI 61508 traite de trois sujets principaux:

1) le cycle de vie de GESTION DES RISQUES et les PROCESSUS de cycle de vie;
2) la définition des niveaux d'intégrité de SECURITE;
3) la recommandation de techniques, d'outils et de méthodes pour le développement de logiciels et les niveaux d'indépendance du personnel chargé d'exécuter les différentes TACHES.

Le point 1) est couvert dans la présente norme par une référence normative à l'ISO 14971 (norme du secteur des DISPOSITIFS MEDICAUX pour la GESTION DES RISQUES). Cette référence a pour effet d'adopter l'approche de l'ISO 14971 en termes de GESTION DES RISQUES comme partie intégrante du PROCESSUS logiciel.

En ce qui concerne le point 2), la présente norme a une approche plus simple que celle de la CEI 61508. Cette dernière classe les logiciels en 4 «Niveaux d'intégrité de SECURITE» définis en termes d'objectifs de fiabilité. Les objectifs de fiabilité sont identifiés après analyse des RISQUES, quantifiant ainsi à la fois la gravité et la probabilité d'un DOMMAGE dû à une défaillance du logiciel.

La présente norme simplifie le point 2) en refusant de tenir compte de la probabilité de défaillance du logiciel avant sa classification. La classification en 3 classes de SECURITE du logiciel est fondée uniquement sur la gravité dudit DOMMAGE causé par une défaillance. Après classification, différents PROCESSUS sont exigés pour les différentes classes de SECURITE de logiciel: l'intention est de réduire encore la probabilité de défaillance du logiciel.

Le point 3) n'est pas traité par la présente norme. Ses utilisateurs sont encouragés à utiliser la CEI 61508 comme source de bonnes méthodes, techniques et outils logiciels tout en reconnaissant que d'autres approches, tant existantes que futures, peuvent fournir des résultats tout aussi bons. La présente norme ne donne pas de recommandations quant à l'indépendance des personnes chargées d'une ACTIVITE logicielle donnée (par exemple la VERIFICATION) par rapport à celles qui sont chargées d'une autre ACTIVITE (par exemple la conception). En particulier, il n'existe pas dans la présente norme d'exigence relative à un évaluateur de SECURITE indépendant car il s'agit d'un sujet couvert par le domaine d'application de l'ISO 14971.

Annexe D
(informative)

Mise en œuvre

D.1 Introduction

La présente annexe présente la manière dont la présente norme peut être mise en œuvre dans les PROCESSUS des FABRICANTS. Elle tient également compte du fait que d'autres normes, telles que l'ISO 13485 [7] exigent des PROCESSUS appropriés et comparables.

D.2 Système de management de la qualité

Pour les FABRICANTs de DISPOSITIFS MEDICAUX, y compris les LOGICIELS DE DISPOSITIFS MEDICAUX dans le contexte de la présente norme, l'établissement d'un système de management de la qualité (SMQ) est exigé en 4.1. La présente norme n'exige pas que le SMQ soit nécessairement certifié.

D.3 ÉVALUATION des PROCESSUS de management de la qualité

Il est recommandé d'évaluer la manière dont les PROCESSUS des SMQ établis et documentés couvrent les PROCESSUS de cycle de vie du logiciel, en réalisant des audits, des inspections ou des analyses sous la responsabilité du FABRICANT. Il peut être remédié à toute carence identifiée en étendant les PROCESSUS des SMQ ou en les décrivant de manière séparée. Si le FABRICANT possède déjà des descriptions de PROCESSUS disponibles qui réglementent le développement, la VERIFICATION et la validation du logiciel, il convient également de les évaluer afin de déterminer leur conformité à la présente norme.

D.4 Intégration des exigences de la présente norme dans les PROCESSUS de management de la qualité des FABRICANTS

La présente norme peut être mise en œuvre en adaptant ou en étendant les PROCESSUS déjà mis en place dans le SMQ ou en intégrant de nouveaux PROCESSUS. La présente norme ne spécifie pas la manière dont cela doit être effectué; le choix de toute méthode convenable est laissé à la discrétion du FABRICANT.

La responsabilité du FABRICANT est de s'assurer que les PROCESSUS décrits dans la présente norme sont correctement mis en application lorsque le logiciel de DISPOSITIF MEDICAL est développé par des équipementiers (OEM) ou des sous-traitants qui ne disposent pas de leur propre SMQ documenté.

D.5 Liste de contrôle pour les petits FABRICANTS ne disposant pas de SMQ certifié

Il convient que le FABRICANT définisse le niveau le plus élevé de classification de SECURITE du logiciel (A, B ou C). Le Tableau D.1 énumère toutes les ACTIVITES décrites dans la présente norme. La référence à l'ISO 13485 a pour intention d'aider à définir la place dans le SMQ. Sur la base de la classe de SECURITE du logiciel exigée, il convient que le FABRICANT évalue chaque ACTIVITE requise au vu des PROCESSUS existants. Si l'exigence est déjà couverte, il est recommandé de renvoyer au descriptif des PROCESSUS pertinents.

En cas de divergence, il est nécessaire de prendre une mesure pour améliorer le PROCESSUS.

Cette liste peut également être utilisée pour une EVALUATION des PROCESSUS une fois la mesure appliquée.

Tableau D.1 – Liste de contrôle pour les petites entreprises sans SMQ certifié

ACTIVITE	Paragraphe correspondant de l'ISO 13485:2003	Couvert par une procédure existante ?	Si oui: Référence	Mesures à prendre
5.1 Planification du développement du logiciel	7.3.1 Planification de la conception et du développement	Oui/Non		
5.2 **Analyses** des exigences du logiciel	7.3.2 Eléments d'entrée de la conception et du développement	Oui/Non		
5.3 Conception ARCHITECTURALE du logiciel		Oui/Non		
5.4 Conception détaillée du logiciel		Oui/Non		
5.5 Mise en œuvre et vérification des UNITES LOGICIELLES		Oui/Non		
5.6 Intégration et essai d'intégration du logiciel		Oui/Non		
5.7 Essais DU SYSTEME LOGICIEL	7.3.3 Résultats en sortie de la conception et du développement 7.3.4 Revue de la conception et du développement	Oui/Non		
5.8 Diffusion du logiciel	7.3.5 Vérification de la conception et du développement 7.3.6 Validation de la conception et du développement	Oui/Non		
6.1 Etablissement du plan de maintenance du logiciel	7.3.7 Maîtrise des modifications de la conception et du développement	Oui/Non		
6.2 Analyse des problèmes et des modifications		Oui/Non		
6.3 Mise en œuvre de la modification	7.3.5 Vérification de la conception et du développement 7.3.6 Validation de la conception et du développement	Oui/Non		
7.1 Analyse du logiciel en termes de contribution à des situations dangereuses		Oui/Non		
7.2 MESURES de MAITRISE DU RISQUE		Oui/Non		
7.3 VERIFICATION des mesures de MAITRISE DU RISQUE		Oui/Non		
7.4 GESTION DES RISQUES des modifications du logiciel		Oui/Non		
8.1 Identification de la configuration	7.5.3 Identification et TRAÇABILITE	Oui/Non		
8.2 Maîtrise des modifications	7.5.3 Identification et TRAÇABILITE	Oui/Non		
8.3 Documentation relative à l'état de la configuration		Oui/Non		
9 PROCESSUS de résolution de problème logiciel		Oui/Non		

Bibliographie

[1] CEI 60601-1:2005, *Appareils électromédicaux – Partie 1: Exigences générales pour la sécurité de base et les performances essentielles*

[2] CEI 60601-1-4:1996, *Appareils électromédicaux – Partie 1-4: Règles générales de sécurité – Norme collatérale: Systèmes électromédicaux programmables*
Amendement 1 (1999)

NOTE Harmonisée comme EN 60601-1-4:1996 + A1:1999 (pas modifiée).

[3] CEI 61508-3, *Sécurité fonctionnelle des systèmes électriques/électroniques/ électroniques programmables relatifs à la sécurité – Partie 3: Prescriptions concernant les logiciels*

NOTE Harmonisée comme EN 61508-3:2001 (pas modifiée).

[4] CEI 61010-1:2001, *Règles de sécurité pour appareils électriques de mesurage, de régulation et de laboratoire – Partie 1: Prescriptions générales*

NOTE Harmonisée comme EN 61010-1:2001 (pas modifiée).

[5] ISO 9000:2005, *Systèmes de management de la qualité – Principes essentiels et vocabulaire*

NOTE Harmonisée comme EN ISO 9000:2005 (pas modifiée).

[6] ISO 9001:2000, *Systèmes de management de la qualité – Exigences*

NOTE Harmonisée comme EN ISO 9001:2000 (pas modifiée).

[7] ISO 13485:2003, *Dispositifs médicaux – Systèmes de management de la qualité – Exigences à des fins réglementaires*

NOTE Harmonisée comme EN ISO 13485:2003 (pas modifiée).

[8] ISO/CEI 9126-1:2001, *Génie du logiciel – Qualité des produits – Partie 1: Modèle de qualité (disponible en anglais seulement)*

[9] ISO/CEI 12207:1995, *Technologies de l'information – Processus du cycle de vie du logiciel (disponible en anglais seulement)*
Amendement 1 (2002)
Amendement 2 (2004)

[10] ISO/CEI 14764:1999, *Technologies de l'information – Maintenance du logiciel (disponible en anglais seulement)*

[11] ISO/CEI 90003:2004, *Ingénierie du logiciel – Lignes directrices pour l'application de l'ISO 9001:2000 aux logiciels informatiques*

[12] ISO/CEI Guide 51:1999, *Aspects liés à la sécurité – Principes directeurs pour les inclure dans les normes*

[13] IEEE 610.12:1990, *Glossaire normalisé IEEE de la terminologie de la technologie de la programmation*

[14] IEEE 1044:1993, *IEEE standard classification for software anomalies*

[15] CEI 60601-1-6, *Appareils électromédicaux - Partie 1-6: Règles générales de sécurité - Norme collatérale: Aptitude à l'utilisation*

NOTE Harmonisée comme EN 60601-1-6:2004 (pas modifiée).

Index des termes définis

Annexe ZA
(normative)

Références normatives à d'autres publications internationales avec les publications européennes correspondantes

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).

NOTE Dans le cas où une publication internationale est modifiée par des modifications communes, indiqué par (mod), l'EN / le HD correspondant(e) s'applique.

Publication	Année	Titre	EN/HD	Année
ISO 14971	-[1)]	Dispositifs médicaux - Application de la gestion des risques aux dispositifs médicaux	EN ISO 14971	2000[2)]

1) Référence non-datée.

2) Edition valide à ce jour.

Annexe ZZ
(informative)

Couverture des Exigences Essentielles des Directives CE

Cette Norme Européenne a été préparée dans le cadre d'un mandat confié au CENELEC par la Commission Européenne et l'Association Européenne de Libre Echange et dans la limite de son domaine d'application la norme couvre les exigences essentielles applicables telles que figurant à l'Annexe I des Directives 93/42/CEE, 90/385/CE et 98/79/CE.
La conformité avec cette norme constitue une méthode de conformité avec les exigences essentielles spécifiées des Directives concernées.
AVERTISSEMENT: D'autres exigences et d'autres Directives CE peuvent être applicables aux produits qui sont couverts par le domaine d'application de cette norme.

_____