



CENTRE NATIONAL D'ÉTUDES SPATIALES

REFERENTIEL NORMATIF du CNES RNC

Référence CNES: **RNC-ECSS-Q-ST-80**

Version 5 du 25 Juin 2010

Référence source : **ECSS-Q-ST-80C**

Version C du 06 Mars 2009

STANDARD

ASSURANCE PRODUIT ASSURANCE PRODUIT LOGICIEL

Traduction française

ACCORD du Bureau de Normalisation	BN n° 19 du 14/11/05 - BN n° 51 du 30/04/09 – BN n° 62 du 14/06/10
APPROBATION IGQ/D Isabelle RONGIER	

PAGE D'ANALYSE DOCUMENTAIRE

TITRE : ASSURANCE PRODUIT LOGICIEL	
MOTS CLES : Assurance produit, logiciel, exigence	
NORME EQUIVALENTE : ECSS-Q-ST-80C	
OBSERVATIONS : Néant	
RESUME : Ce document contient les exigences qualité à respecter pour le développement et la maintenance des logiciels des systèmes spatiaux.	
SITUATION DU DOCUMENT : Ce document fait partie de la collection des standards associés au domaine « Assurance Produit » du Référentiel Normatif du CNES.	
NOMBRE DE PAGES : 124	LANGUE : Française
Progiciels utilisés / version : Word 2007	
SERVICE GESTIONNAIRE : Inspection Générale Direction de la Fonction Qualité (IGQ)	

SUIVI DES VERSIONS DU DOCUMENT RNC

VERSION	DATE	PAGES MODIFIEES	OBSERVATIONS
1	18/05/2002		Introduction dans le RNC de la version A du document ECSS-Q-80.
2	05/07/2004		Suite au BN n° 5 introduction dans le RNC de la version B du document ECSS-Q-80.
3	14/12/2005		Rajout de modalités d'application, suite au problème de numérotation des paragraphes de l' ECSS-Q-00 lors du passage en EN (RNC-CEN-Q-00), acceptées au BN n° 19 du 14/11/2005 (Cf. FEB 33/2005).
4	05/05/2009	Toutes	Suite à la revue de cohérence associée à la phase de benchmarking ECSS, changement de nomenclature et introduction dans le RNC (BN n° 46 du 24/11/2008) de la version C du document ECSS-Q-ST-80.
5	25/05/2010		Mise en cohérence de la partie termes, définitions et abréviations avec la traduction française de l'ECSS-E-ST-40. Les corrections issues des modalités d'application introduites dans la version anglaise (version 5) de ce document, ont été intégrées directement dans les chapitres correspondants.

MODALITES PARTICULIERES D'APPLICATION

Pas de modalités particulières



Assurance produit des projets spatiaux

Assurance produit logiciel

Avant-propos

Le présent standard fait partie de la série des standards ECSS destinés à être appliqués au management, à l'ingénierie et à l'assurance produit dans les projets spatiaux et dans les applications spatiales. L'ECSS, coopération européenne pour la normalisation spatiale (European Cooperation for Space Standards) est l'aboutissement des efforts communs de l'Agence Spatiale Européenne, des agences spatiales nationales et des associations industrielles européennes travaillant au développement et au maintien de standards communs. Les exigences du présent standard sont exprimées en termes d'objectifs à réaliser, plutôt qu'en fonction de la façon d'organiser et d'effectuer le travail nécessaire. Les structures et les méthodes d'organisation existantes peuvent ainsi être appliquées lorsqu'elles sont efficaces et évoluer, si nécessaire, sans que les standards n'aient besoin d'être modifiés.

Le présent standard a été préparé par le Groupe de Travail ECSS-Q-ST-80C de l'ECSS, revue par le Secrétariat exécutif de l'ECSS et approuvée par l'Autorité technique de l'ECSS.

Décharge de responsabilité

L'ECSS n'offre aucune garantie, de quelque nature que ce soit, qu'elle soit expresse, implicite ou réglementaire, y compris, mais sans y être limité, toute garantie d'aptitude à la vente ou à un usage particulier, ou toute garantie que le contenu de l'article est sans défaut. L'ECSS ne saurait en aucun cas être tenu pour responsable de tout dommage, y compris, mais sans y être limité, les dommages directs, indirects, spécifiques ou consécutifs, provoqués par, résultant de, ou liés de quelque manière que ce soit à l'utilisation de ce standard, que ce soit sur la base d'une garantie, d'un accord commercial, d'un délit, ou autre ; qu'une atteinte ait pu être subie ou non par des personnes ou des biens ou autre ; et qu'une perte ait pu être causée par, ou résulter de l'article, ou de toute prestation qui aurait pu être réalisée par l'ECSS.

Publié par : ESA Division des Normes et Exigences
ESTEC, P.O. Box 299,
2200 AG Noordwijk
Pays-Bas

Copyright : 2009 © Agence Spatiale Européenne au profit des membres de l'ECSS

Historique des modifications

ECSS-Q-80A 19 avril 1996	Première édition
ECSS-Q-80B 10 octobre 2003	Deuxième édition
ECSS-Q-ST-80C 6 mars 2009	Troisième édition Les principales modifications par rapport à la version précédente sont les suivantes : <ul style="list-style-type: none">• définition des catégories de criticité des logiciels et adaptation du standard en fonction de ces catégories ;• amélioration des exigences sur la réutilisation, la sécurité et la sûreté de fonctionnement des logiciels, et l'évaluation et l'amélioration des processus logiciel ;• simplification des exigences en vue d'une meilleure adaptation du standard à une utilisation directe dans les accords commerciaux.

Table des matières

Historique des modifications	4
1 Domaine d'application	10
2 Références normatives	11
3 Termes, définitions et abréviations	12
3.1 Termes définis dans d'autres standards.....	12
3.2 Termes spécifiques au présent standard.....	12
3.3 Abréviations.....	17
4 Principes d'assurance produit logiciel des systèmes spatiaux.....	21
4.1 Introduction.....	21
4.2 Organisation du présent standard	22
4.3 Adaptation du présent standard.....	24
5 Mise en œuvre du programme d'assurance produit logiciel.....	25
5.1 Organisation et responsabilités	25
5.1.1 Organisation	25
5.1.2 Responsabilités et autorité.....	25
5.1.3 Ressources	26
5.1.4 Responsable/ingénieur assurance produit logiciel	26
5.1.5 Formation	26
5.2 Gestion du programme d'assurance produit logiciel.....	27
5.2.1 Planification et contrôle de l'assurance produit logiciel.....	27
5.2.2 Bilans qualité logiciel.....	28
5.2.3 Audits	29
5.2.4 Alertes	29
5.2.5 Problèmes logiciel	29
5.2.6 Non-conformités.....	30

5.2.7	Exigences qualité et modèles qualité	30
5.3	Management des risques et maîtrise des éléments critiques	31
5.3.1	Management des risques	31
5.3.2	Maîtrise des éléments critiques	31
5.4	Sélection et suivi des fournisseurs	32
5.4.1	Sélection des fournisseurs	32
5.4.2	Exigences pour les fournisseurs	32
5.4.3	Surveillance des fournisseurs	32
5.4.4	Classification de la criticité	33
5.5	Approvisionnement	33
5.5.1	Documents d'approvisionnement.....	33
5.5.2	Revue de la liste des composants logiciel acquis.....	33
5.5.3	Caractérisation des approvisionnements	34
5.5.4	Identification	34
5.5.5	Contrôle.....	34
5.5.6	Export.....	34
5.6	Outils et environnement support.....	34
5.6.1	Méthodes et outils	34
5.6.2	Sélection de l'environnement de développement.....	35
5.7	Processus d'évaluation et d'amélioration	36
5.7.1	Evaluation des processus.....	36
5.7.2	Processus d'évaluation	36
5.7.3	Amélioration des processus.....	37
6 Assurance des processus logiciel		39
6.1	Cycle de vie du développement logiciel.....	39
6.1.1	Définition du cycle de vie	39
6.1.2	Objectifs qualité des processus	39
6.1.3	Revue de la définition du cycle de vie	39
6.1.4	Ressources du cycle de vie.....	40
6.1.5	Planification du processus de validation logiciel	40
6.2	Exigences applicables à tous les processus d'ingénierie logiciel	40
6.2.1	Documentation des processus	40
6.2.2	Sûreté de fonctionnement des logiciels et sécurité.....	41
6.2.3	Traitement des logiciels critiques.....	43
6.2.4	Gestion de configuration des logiciels	46
6.2.5	Métriques sur les processus.....	48
6.2.6	Vérification.....	49
6.2.7	Réutilisation de logiciels existants	51
6.2.8	Génération automatique de codes	55
6.3	Exigences applicables à chacun des processus ou activités d'ingénierie logiciel	56
6.3.1	Traitement des exigences système relatives au logiciel.....	56
6.3.2	Analyse des exigences logiciel.....	56
6.3.3	Conception de l'architecture du logiciel et des éléments logiciel.....	58
6.3.4	Codage.....	59
6.3.5	Tests et validation.....	60

6.3.6	Livraison et recette des logiciels	65
6.3.7	Opérations	67
6.3.8	Maintenance	67
7 Assurance qualité du produit logiciel		70
7.1	Objectifs qualité produit et métriques	70
7.1.1	Déclinaison des exigences.....	70
7.1.2	Définition quantitative des exigences qualité.....	70
7.1.3	Activités d'assurance relatives aux exigences qualité produit.....	70
7.1.4	Métriques sur les produits	70
7.1.5	Métriques de base.....	71
7.1.6	Rapports sur les métriques	71
7.1.7	Précision numérique	71
7.1.8	Analyse de la maturité du logiciel.....	72
7.2	Exigences qualité produit.....	72
7.2.1	Exigences de besoin et spécification technique	72
7.2.2	Conception et documentation associée	72
7.2.3	Documentation de test et de validation	73
7.3	Logiciels destinés à être réutilisés	74
7.3.1	Exigences du client.....	74
7.3.2	Documentation séparée	74
7.3.3	Informations autosuffisantes	74
7.3.4	Exigences relatives aux réutilisations prévues	74
7.3.5	Gestion de configuration en vue d'une réutilisation	74
7.3.6	Tests sur plusieurs plates-formes	75
7.3.7	Certificat de conformité	75
7.4	Matériel et services au sol standard pour un système opérationnel.....	75
7.4.1	Acquisition de matériel	75
7.4.2	Acquisition de services.....	75
7.4.3	Contraintes	76
7.4.4	Sélection	76
7.4.5	Maintenance	76
7.5	Microprogrammes	76
7.5.1	Programmation des dispositifs	76
7.5.2	Marquage.....	77
7.5.3	Etalonnage.....	77
Annexe A (informative) Documentation logiciel.....		78
Annexe B (normative) Plan assurance produit logiciel (SPAP) - DRD.....		84
B.1	Identification de la DRD.....	84
B.2	Réponse attendue	85

Annexe C (normative) Bilans qualité logiciel de jalon (SPAMR) - DRD 90

C.1	Identification de la DRD.....	90
C.2	Réponse attendue.....	91

Annexe D (normative) Adaptation du standard en fonction de la criticité des logiciels 93

D.1	Catégories de criticité des logiciels.....	93
D.2	Matrice d'applicabilité.....	94

Annexe E (informative) Liste des exigences adaptables 105**Annexe F (informative) Organisation et contenu des documents à chaque jalon 106**

F.1	Introduction.....	106
F.2	Résultats attendus de l'ECSS-Q-ST-80 - SRR.....	106
F.3	Résultats attendus de l'ECSS-Q-ST-80 - PDR.....	108
F.4	Résultats attendus de l'ECSS-Q-ST-80 - CDR.....	109
F.5	Résultats attendus de l'ECSS-Q-ST-80 - QR.....	109
F.6	Résultats attendus de l'ECSS-Q-ST-80 - AR.....	109
F.7	Résultats attendus de l'ECSS-Q-ST-80 non associés à des revues de jalon spécifiques.....	109

Bibliographie..... 109**Figures**

Figure 4-1	: Processus logiciel dans les standards ECSS.....	22
Figure 4-2	: Structure du présent standard.....	23
Figure A-1	: Présentation générale des documents logiciel.....	78

Tableaux

Tableau A-1 : Liste des exigences documentaires (SRF) des standards ECSS-E-ST-40 et ECSS-Q-ST-8079

Tableau B-1 : Traçabilité du SPAP par rapport aux paragraphes des standards ECSS-E-ST-40 et ECSS-Q-ST-80.....84

Tableau C-1 : Traçabilité du SPAP par rapport aux paragraphes des standards ECSS-E-ST-40 et ECSS-Q-ST-80.....90

Tableau D-1 : Catégories de criticité des logiciels.....93

Tableau D-2 : Matrice d'applicabilité en fonction de la criticité des logiciels94

1

Domaine d'application

Le présent standard définit un ensemble d'exigences d'assurance produit logiciel à utiliser pour le développement et la maintenance des logiciels des systèmes spatiaux. Les systèmes spatiaux comprennent les véhicules spatiaux automatisés ou à pilotage manuel, les lanceurs, les charges utiles, les expériences et leurs équipements et moyens sol associés. Les logiciels incluent les composants logiciel intégrés ou microprogrammes.

Ce standard s'applique aussi au développement ou à la réutilisation de logiciels non livrables qui affectent la qualité du produit ou service livrable fourni par un système spatial, si le service est rempli par des logiciels.

Le standard ECSS-Q-ST-80 interagit avec l'ingénierie et le management des produits spatiaux, traités dans les branches Ingénierie (-E) et Management (-M) du système ECSS, et explique comment elles sont liées aux processus d'assurance produit logiciel.

Le présent standard peut être adapté aux caractéristiques et contraintes spécifiques à un projet spatial, selon l'ECSS-S-ST-00.

L'adaptation de ce standard à un accord commercial ou projet spécifique lors de la préparation des exigences d'assurance produit logiciel est définie dans le paragraphe 4.3.

2**Références normatives**

Les documents normatifs suivants contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour le présent standard ECSS. Pour les références datées, les amendements ultérieurs ou les révisions de ces publications ne s'appliquent pas. Toutefois, les parties prenantes aux accords fondés sur le présent standard ECSS sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des documents normatifs indiqués ci-après. Pour les références non datées, la dernière édition du document en référence s'applique.

ECSS-S-ST-00-01	Système ECCS – Glossaire des termes
ECSS-E-ST-40	Ingénierie spatiale – Exigences générales de logiciel
ECSS-Q-ST-10	Assurance Produit des projets spatiaux – Management de l'assurance produit
ECSS-Q-ST-10-04	Assurance Produit des projets spatiaux – Maîtrise des éléments critiques
ECSS-Q-ST-10-09	Assurance produit des projets spatiaux – Instruction et traitement des anomalies
ECSS-Q-ST-20	Assurance produit des projets spatiaux – Assurance qualité
ECSS-Q-ST-30	Assurance produit des projets spatiaux – Sûreté de fonctionnement
ECSS-Q-ST-40	Assurance produit des projets spatiaux – Sécurité
ECSS-M-ST-10	Management des projets spatiaux – Planification et mise en œuvre de projet
ECSS-M-ST-10-01	Management des projets spatiaux – Organisation et conduite des revues
ECSS-M-ST-40	Management des projets spatiaux – Gestion de configuration et des informations
ECSS-M-ST-80	Management des projets spatiaux – Management des risques
ISO/IEC 15504 Partie 2 :2003	Ingénierie du logiciel – Evaluation des procédés – Partie 2 : Conseils sur la réalisation d'une évaluation – Première édition

Termes, définitions et abréviations

3.1 Termes définis dans d'autres standards

Pour les besoins du présent standard, les termes et définitions donnés dans l'ECSS-ST-00-01 s'appliquent, notamment pour les termes suivants :

Essai de recette

Produit logiciel

NOTE Les termes et définitions sont les mêmes pour les standards ECSS-E-ST-40 et ECSS-Q-ST-80.

3.2 Termes spécifiques au présent standard

3.2.1 Assurance produit logiciel

Ensemble des activités, standards, contrôles et procédures réalisés tout au long de la durée de vie d'un produit logiciel, qui permet de garantir que le produit logiciel fourni, ou le logiciel influant sur la qualité du produit fourni, est conforme aux exigences du client

3.2.2 Caractéristiques qualité (logiciel)

Ensemble d'attributs d'un produit logiciel au travers desquels sa qualité est décrite et évaluée

NOTE Chaque caractéristique qualité d'un logiciel peut posséder plusieurs sous caractéristiques.

3.2.3 Cas test

Ensemble d'entrées, de conditions d'exécution et de résultats attendus pour un test donné, définis en vue d'atteindre un objectif spécifique (par exemple, suivre un programme donné ou vérifier la conformité à une exigence spécifiée).

3.2.4 Code configurable

Code (code source ou code exécutable) qui peut être adapté par la modification des valeurs de certains paramètres.

NOTE Cette définition couvre notamment les classes de code configurable obtenues par les moyens de configuration suivants :

- configuration fondée sur l'utilisation d'une directive de compilation ;
- configuration fondée sur l'utilisation d'une directive d'éditeur de lien;
- configuration réalisée via un paramètre défini dans un fichier de configuration ;
- configuration réalisée via des données définies dans une base de données ayant un impact sur les parties réellement exécutables du logiciel (par exemple, les paramètres définissant les structures de branche qui entraînent la non-exécution de certaines parties existantes du code).

3.2.5 Code désactivé

Code qui, bien qu'intégré correctement dans la conception et le codage, n'est destiné à être exécuté que dans certaines configurations du produit logiciel ou dans aucune d'entre elles.

[adapté de RTCA/DO-178B]

3.2.6 Code mort

Code qui ne peut pas être exécuté du fait d'une erreur de conception ou de codage.

3.2.7 Composant logiciel

Partie d'un système logiciel

NOTE 1 Composant logiciel est utilisé comme terme général.

NOTE 2 Les composants peuvent être assemblés et décomposés pour former de nouveaux composants. Dans les activités de production, les composants sont mis en œuvre sous la forme d'unités, de tâches ou de programmes, chacun pouvant constituer un article de configuration. Cette utilisation du terme est plus générale que dans la terminologie ANSI/IEEE, qui définit un composant comme une « pièce de base d'un système ou programme » ; dans le présent standard, les composants ne sont pas toujours « de base » puisqu'ils peuvent être décomposés.

3.2.8 Conception de test

Documentation qui décrit la démarche de test associée à une fonctionnalité ou caractéristique logiciel ou à une combinaison de fonctionnalités ou caractéristiques logiciel, et qui identifie les tests associés.

3.2.9 Condition

Expression booléenne ne comportant pas d'opérateur booléen

3.2.10 Couverture des décisions et conditions modifiées

Mesure de la partie du programme dans laquelle chaque point d'entrée et de sortie a été invoqué au moins une fois, chaque décision du programme a pris la valeur « vrai » et « faux » au moins une fois et chaque condition d'une décision s'est avérée avoir une incidence propre sur le résultat de cette décision

NOTE Pour montrer qu'une condition a une incidence propre sur le résultat d'une décision, il suffit de faire varier la valeur de cette condition tout en conservant celle des autres conditions possibles.

3.2.11 Examen d'ensemble

Technique d'analyse statique par laquelle un concepteur ou un programmeur présente le produit logiciel aux membres de l'équipe de développement et aux autres parties intéressées et les participants posent des questions et font des commentaires sur les erreurs possibles, la violation des normes de développement et d'autres problèmes.

[IEEE 1028-1997]

3.2.12 Couverture des décisions

Mesure d'une partie du programme dans laquelle chaque point d'entrée et de sortie est invoqué au moins une fois et chaque décision prend les valeurs « vrai » et « faux » au moins une fois.

NOTE La couverture des décisions comprend par définition la couverture des instructions.

3.2.13 Couverture des instructions

Mesure de la partie du programme dans laquelle chaque instruction de code source exécutable a été invoquée au moins une fois.

3.2.14 Couverture du code

Pourcentage du logiciel exécuté (couvert) par la série de test.

3.2.15 Décision

Expression booléenne composée de conditions et de zéro ou plusieurs opérateurs booléens utilisés dans une structure de contrôle

NOTE 1 Par exemple : L'instruction « if.....thenelse » ou « case » constitue une structure de contrôle.

NOTE 2 Une décision sans opérateur booléen est une condition.

NOTE 3 Si une condition apparaît plusieurs fois dans une décision, chaque occurrence constitue une condition distincte.

3.2.16 Élément logiciel

Voir « produit logiciel » dans l'ECSS-S-ST-00-01

3.2.17 Entrée singulière

Entrée correspondant à une singularité de la fonction

3.2.18 Essais de non régression (logiciel)

Rejeu sélectif d'essais d'un système ou d'un composant pour vérifier que des modifications n'ont pas causé d'effets indésirables et que le système ou le composant satisfait toujours aux exigences spécifiées

[IEEE 610.12:1990]

3.2.19 Essais d'intégration

Essais lors desquels des composants logiciel et/ou matériel sont assemblés et testés pour évaluer leur interaction réciproque

[IEEE 610.12:1990]

3.2.20 Evalueur compétent

Personne qui possède les connaissances, les compétences et l'expérience nécessaires pour mener une évaluation des processus conformément à la norme ISO/IEC 15504

NOTE Adapté de la norme ISO/IEC 15504:1998,
Partie 9.

3.2.21 Facilité d'utilisation (caractéristique qualité)

Aptitude d'un logiciel à être compris, appris, utilisé et apprécié de l'utilisateur, lorsqu'il est utilisé dans les conditions spécifiées.

3.2.22 Génération automatique de codes

Génération de code source à l'aide d'un outil à partir d'un modèle

3.2.23 Logiciel

Voir « produit logiciel » dans l'ECSS-S-ST-00-01

3.2.24 Logiciel critique

Logiciel appartenant à la catégorie de criticité A, B ou C

NOTE Voir le Tableau D-1 – Catégories de criticité des logiciels de l'ECSS-Q-ST-80C.

3.2.25 Logiciel de type COTS, OTS, MOTS

Dans le cadre du présent standard, produits sur étagère, logiciels sur étagère et logiciels sur étagère modifiés pour lesquels il existe une preuve d'utilisation.

3.2.26 Logiciel existant

Tout logiciel développé hors du cadre de l'accord commercial auquel ce standard s'applique, y compris les logiciels issus de précédents développements transmis par le fournisseur ou le client, les logiciels de type COTS, OTS et MOTS, les logiciels gratuits et les logiciels libres

3.2.27 Métrique

Méthode de mesure définie et échelle de mesure associée.

NOTE 1 Les métriques peuvent être internes ou externes, directes ou indirectes.

NOTE 2 Les métriques incluent les méthodes de caractérisation des données qualitatives.

[ISO/IEC 9126-1:2001]

3.2.28 Migration

Portage d'un produit logiciel dans un nouvel environnement

3.2.29 Modèle logique

Modèle d'éléments logiciel, indépendant de la mise en œuvre, utilisé pour analyser et documenter les exigences logiciel

3.2.30 Modèle qualité (logiciel)

Ensemble de caractéristiques et de leurs relations qui fournit les bases pour la spécification des exigences qualité et l'évaluation de la qualité

[ISO/IEC 9126-1:2001]

3.2.31 Observabilité du logiciel

Propriété d'un système pour lequel la valeur des variables d'état peut être déterminée grâce à l'observation des variables en sortie

3.2.32 Opérationnel

Dans le cadre du présent standard, relatif à l'exploitation des logiciels

NOTE Ce terme ne fait pas référence à l'exploitation du véhicule spatial.

3.2.33 Philosophie des marges

Analyse raisonnée pour les marges allouées aux paramètres de performance et aux ressources machines d'un développement et manière dont ces marges doivent être gérées pendant l'exécution du projet.

3.2.34 Portabilité (caractéristique qualité)

Capacité d'un logiciel à être transféré d'un environnement à un autre.

3.2.35 Problème logiciel

Condition d'un produit logiciel susceptible d'être la source de difficultés ou d'incertitudes lors de son utilisation

[CMU/SEI-92-TR-022]

3.2.36 Procédure de test

Instructions détaillées relatives à l'organisation, à l'utilisation et à l'évaluation des résultats d'un test donné

3.2.37 Produits de mission

Produits et services fournis par le système spatial

NOTE Par exemple : services de communication, données scientifiques.

3.2.38 Réutilisabilité

Capacité d'une unité logiciel ou d'un autre produit de développement à être utilisé dans plus d'un programme informatique ou système logiciel

[IEEE 610.12:1990]

3.2.39 Script de test

Fichier contenant un ensemble de commandes ou d'instructions écrites au format natif (traitement par ordinateur ou outil) dans le but d'automatiser l'exécution d'une ou plusieurs procédures de test (ainsi que l'évaluation des résultats correspondants)

3.2.40 Système à logiciel prépondérant

Système spatial dans lequel la majeure partie des constituants sont des éléments logiciel

NOTE Dans de tels systèmes, les sous-systèmes sont essentiellement constitués de logiciels. Pour ce type de système, la majorité des interfaces sont des interfaces logiciel-logiciel.

3.2.41 Temps réel

Appartenance à un système ou à un mode d'exploitation dans lequel les calculs sont réalisés au moment même où un processus extérieur se déroule, de sorte que leurs résultats puissent être utilisés pour contrôler, surveiller ou répondre en temps voulu au processus extérieur

[IEEE 610.12:1990]

3.2.42 Test aux limites

Test qui évalue un système ou un composant logiciel au niveau ou au-delà de ses capacités requises.

3.2.43 Test unitaire

Essai d'une unité logiciel individuelle

3.2.44 Unité logiciel

Élément de code source qui peut être compilé séparément.

NOTE Dans le présent standard, aucune différence n'est faite entre une unité logiciel et une base de données ; les deux sont couvertes par les mêmes exigences.

3.2.45 Validation

Processus <logiciel> qui permet de confirmer que les fonctions de base et les performances spécifiées dans les exigences de besoin sont correctes et entièrement mises en œuvre dans le produit final.

3.2.46 Vérification

Processus <logiciel> qui permet de confirmer que les spécifications et les entrées adéquates existent pour toutes les activités et que les résultats de ces activités sont corrects et cohérents par rapport aux spécifications et aux entrées.

3.3 Abréviations

Pour les besoins du présent standard et de l'ECSS-E-ST-40, les abréviations données dans l'ECSS-S-ST-00-01 s'appliquent, ainsi que les suivantes :

Pour obtenir la définition des acronymes des DRD, voir l'Annexe A.

NOTE Les abréviations sont les mêmes pour les standards ECSS-E-ST-40 et ECSS-Q-ST-80.

Abréviation	Signification
AR	<i>Acceptance Review</i> revue de recette NOTE Le terme SW-AR peut être utilisé pour désigner clairement les AR concernant uniquement des produits logiciel.
CDR	<i>Critical Design Review</i> revue critique de définition NOTE Le terme SW-CDR peut être utilisé pour désigner clairement les CDR concernant uniquement des produits logiciel.
CMMI	<i>Capability Maturity Model Integration</i> <i>intégration d'un modèle de maturité</i>
COTS	<i>Commercial Off-The-Shelf</i> produits commerciaux sur étagère
CPU	<i>unité centrale Central Processing Unit</i> unité centrale de traitement (processeur)
DDF	<i>Design Definition File</i> dossier de conception
DDR	<i>Detailed Design Review</i> revue de conception détaillée
DJF	<i>Design Justification File</i> dossier justificatif de définition
DRD	<i>Document Requirements Definition</i> définition des exigences documentaires
ECSS	<i>European Cooperation for Space Standardization</i> Ccoopération européenne pour la normalisation spatiale
EO	<i>Expected Output</i> résultat attendu
GS	<i>Ground Segment</i> segment sol
HMI	<i>Human-Machine Interface</i> interface homme machine
HSIA	<i>Hardware-Software Interaction Analysis</i> analyse des interactions matériel logiciel
HW	<i>HardWare</i> matériel
ICD	<i>Interface Control Document</i>

	document de contrôle des interfaces
INTRSA	<i>InternatioNal regisTRation Scheme for Assessors</i> système d'inscription international des évaluateurs
IRD	<i>Interface Requirements Document</i> spécification de gestion des interfaces
ISO	<i>International Standardization Organization</i> organisation internationale de normalisation
ISV	<i>Independent Software Validation</i> validation indépendante des logiciels
ISVV	<i>Independent Software Verification and Validation</i> vérification et validation indépendantes des logiciels
MF	<i>Maintenance File</i> dossier de maintenance suiveur
MGT	<i>Management File</i> dossier de gestion
MOTS	<i>Modified Off-The-Shelf</i> produits sur étagère modifiés
OBCP	<i>On-Board Control Procedure</i> procédure de commande de bord
OP	<i>Operational Plan</i> plan d'exploitation
ORR	<i>Operational Readiness Review</i> revue d'aptitude opérationnelle
OTS	<i>Off-The-Shelf</i> produits sur étagère
PAF	<i>Product Assurance File</i> dossier d'assurance produit
PDR	<i>Preliminary Design Review</i> revue de conception préliminaire NOTE Le terme SW-PDR peut être utilisé pour désigner clairement les PDR concernant uniquement des produits logiciel.
PRR	<i>Preliminary Requirement Review</i> revue des exigences préliminaires
QR	<i>Qualification Review</i> revue de qualification NOTE Le terme SW-QR peut être utilisé pour désigner clairement les QR concernant uniquement des produits logiciel.
RB	<i>Requirements Baseline</i> exigences de besoin
SCAMPI	<i>Standard CMMI Appraisal Method for Process Improvement</i> méthode d'évaluation CMMI standard pour l'amélioration des processus

SDE	<i>Software Development Environment</i> environnement de développement logiciel
SOS	<i>Software Operation Support</i> support à l'exploitation des logiciels
SPA	<i>Software Product Assurance</i> assurance produit logiciel
SPAMR	<i>Software Product Assurance Milestone Report</i> bilan qualité logiciel de jalon
SPAP	<i>Software Product Assurance Plan</i> plan assurance produit logiciel
SPR	<i>Software Problem Report</i> fiche de problème logiciel
SRB	<i>Software Review Board</i> commission de revue du logiciel
SRR	<i>System Requirements Review</i> revue des exigences système ou revue d'exigences de besoin ou revue de définition préliminaire NOTE Le terme SW-SRR peut être utilisé pour désigner clairement les SRR concernant uniquement des produits logiciel.
SW	<i>SoftWare</i> logiciel
SWE	<i>Software Engineering</i> ingénierie logiciel
TRR	<i>Test Readiness Review</i> revue d'aptitude au test
TS	<i>Technical Specification</i> spécification technique

4

Principes d'assurance produit logiciel des systèmes spatiaux

4.1 Introduction

L'assurance produit logiciel a pour objectif de garantir au client et au fournisseur que les logiciels développés ou acquis/réutilisés satisfont à ses exigences tout au long de la durée de vie du système. Elle vise notamment à garantir que les logiciels sont développés de manière à fonctionner correctement et sûrement dans leur environnement d'exploitation, en respectant les objectifs de qualité définis pour le projet.

Le présent standard contribue à ces objectifs en définissant les exigences d'assurance produit logiciel à respecter dans un projet spatial donné. Ces exigences traitent de la gestion et de la structure qualité, de la définition des activités et des processus du cycle de vie et des caractéristiques qualité des produits.

L'un des principes fondamentaux du présent standard est la relation client-fournisseur, supposée pour tous les développements logiciel. Les aspects organisationnels de cette relation sont définis dans l'ECSS-M-ST-10. Le client est en général l'acquéreur de deux produits fortement imbriqués : les composants matériel et logiciel d'un système, sous-système, ensemble, équipement ou assemblage. Le concept de relation client-fournisseur est appliqué de manière récursive, c'est-à-dire que le client peut lui-même être le fournisseur d'un niveau supérieur de la hiérarchie du système spatial.

Sauf indication contraire, les exigences du présent standard s'appliquent au fournisseur.

Le fournisseur démontre la conformité aux exigences d'assurance produit logiciel et fournit la preuve de conformité spécifiée.

A cet effet, le fournisseur spécifie les exigences d'assurance produit logiciel destinées à ses fournisseurs en tenant compte de leurs responsabilités et de la nature particulière de leurs livraisons.

Le présent standard complète l'ECSS-E-ST-40 « Ingénierie spatiale – Exigences générales logiciel » ; elle introduit les aspects d'assurance produit, intégrés dans les processus d'ingénierie logiciel des systèmes spatiaux, tels que définis dans l'ECSS-E-ST-40. Ces deux standards définissent l'ensemble des processus de développement de logiciels spatiaux.

La Figure 4-1 présente, sous forme de schéma, les différents processus logiciel traités par l'ensemble des standards ECSS.

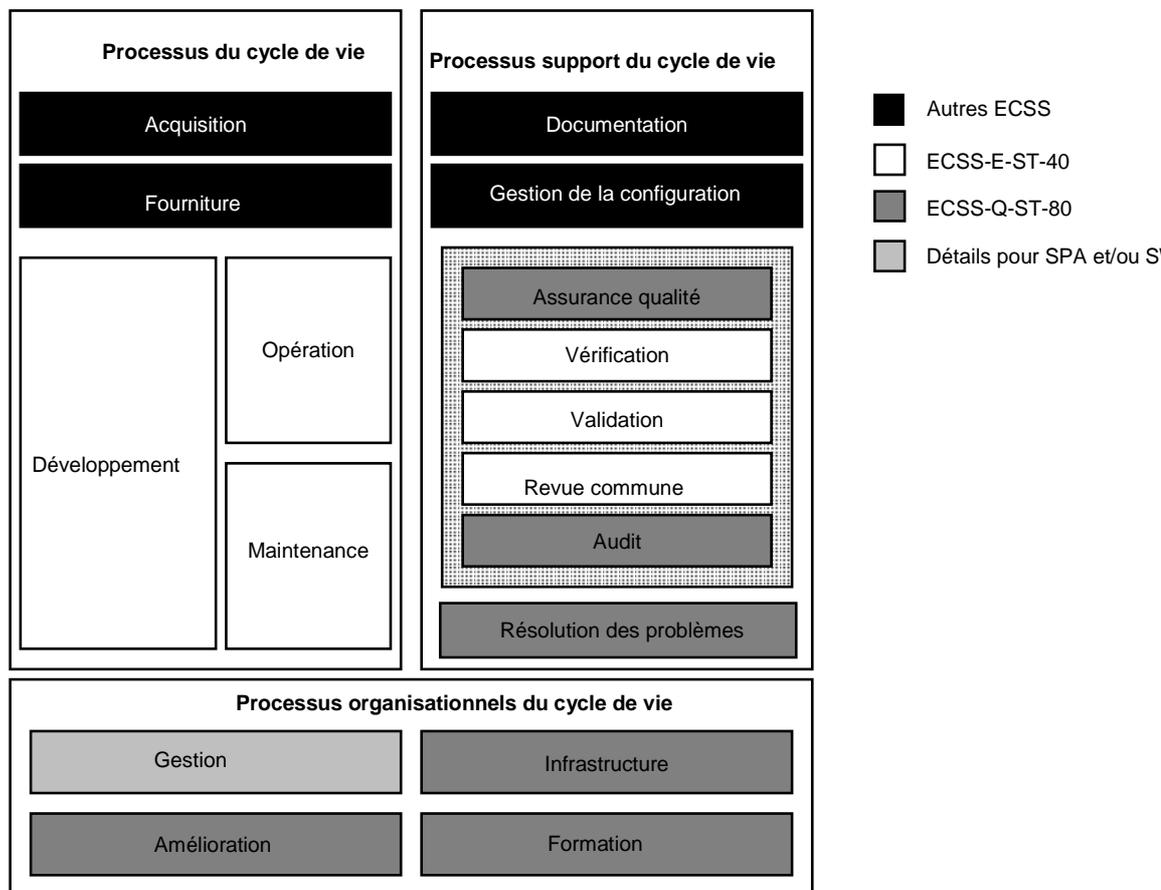


Figure 4-1 : Processus logiciel dans les standards ECSS

4.2 Organisation du présent standard

Le présent standard est organisé en trois parties principales :

- mise en œuvre du programme d'assurance produit logiciel ;
- assurance qualité des processus logiciel ;
- assurance qualité du produit logiciel.
-

La documentation logiciel, qui réunit les résultats attendus des exigences de l'ECSS-E-ST-40 et de l'ECSS-Q-ST-80, est résumée à l'Annexe A.

L'Annexe B et l'Annexe C définissent les DRD (définition des exigences documentaires) des documents d'assurance produit logiciel (SPAP et SPAMR). Les DRD des autres documents d'ingénierie logiciel et de gestion sont inclus dans l'ECSS-E-ST-40 et l'ECSS-M-ST-40.

Dans la préparation du présent standard, la norme ISO/IEC 12207 a été largement utilisée, fournissant une trame commune reconnue internationalement pour la terminologie et la description des processus du cycle de vie logiciel.

L'organisation du présent standard est détaillée dans la Figure 4-2.

Mise en œuvre du programme d'assurance produit logiciel	
5.1 Organisation et responsabilités	5.5 Approvisionnement
5.2 Gestion du programme d'assurance produit logiciel	5.6 Outils et environnement support
5.3 Management des risques et maîtrise des éléments critiques	5.7 Processus d'évaluation et d'amélioration
5.4 Sélection et suivi des fournisseurs	

Assurance des processus logiciel
6.1 Cycle de vie du développement logiciel
6.2 Exigences applicables à tous les processus d'ingénierie logiciel
6.2.8.7 La documentation de vérification et de validation doit distinguer les activités à réaliser pour le code généré manuellement des activités à réaliser pour le code généré de manière automatique. <i>RESULTAT ATTENDU : documentation de validation et des essais [DJF, SValP; RDP], [DJF, SVS; RCD, RQ, RA], [DJF, SUITP; RDP, RCD].</i>
Exigences applicables à chacun des processus ou activités d'ingénierie logiciel

Assurance qualité du produit logiciel
7.1 Objectifs qualité produit et métriques
7.2 Exigences qualité produit
7.3 Logiciels destinés à être réutilisés
7.4 Matériel et services au sol standard pour un système opérationnel
7.5 Microprogrammes

Figure 4-2 : Structure du présent standard

Chaque exigence du présent standard est identifiée par un numéro hiérarchisé suivi d'une lettre, le cas échéant (par exemple, 5.3.1.5, point a). Les résultats

associés à chaque exigence sont donnés dans la section « Résultat attendu ». Lorsque plusieurs résultats sont attendus, ils sont identifiés par une lettre (par exemple, « a », « b », etc.). La destination de chaque résultat est indiquée entre crochets, avec la DRD correspondantes (après une virgule) et la ou les revues (après un point-virgule). Par exemple : « [PAF, SPAP; SRR] » annonce un résultat contenu dans le plan assurance produit logiciel, faisant partie du dossier d'assurance produit et requis pour la SRR. En l'absence de DRD pour un résultat attendu et/ou si le résultat attendu ne doit pas être fourni lors d'une revue spécifique, les sections correspondantes de ce résultat sont remplacées par des tirets (par exemple, « [PAF, -; -] »).

Le présent standard précise pour les aspects assurance produit logiciel certaines des exigences générales déjà traitées dans les standards ECSS de management, d'assurance produit et d'assurance qualité.

4.3 Adaptation du présent standard

Les informations et exigences générales relatives à la sélection et à l'adaptation des standards applicables sont définies dans l'ECSS-S-ST-00.

Plusieurs éléments peuvent piloter l'adaptation, et notamment les aspects relatifs à la sûreté de fonctionnement et à la sécurité, les contraintes de développement logiciel, les objectifs de qualité du produit et les objectifs économiques.

L'adaptation par rapport aux aspects relatifs à la sûreté de fonctionnement et à la sécurité repose sur la sélection des exigences liées à la vérification, la validation et aux niveaux de preuve exigés par la criticité du logiciel. L'Annexe D offre une adaptation du présent standard en fonction de la criticité des logiciels.

L'adaptation par rapport aux contraintes de développement logiciel prend en compte les caractéristiques spécifiques des logiciels à développer et de l'environnement de développement. Le type de développement logiciel (par exemple, base de données ou temps réel) et le système cible (par exemple, processeur encapsulé, système hôte, dispositifs programmables ou circuits intégrés spécifiques à une application) sont également pris en compte (voir l'Annexe S de l'ECSS-E-ST-40). Des exigences particulières en matière de vérification, de revue et de contrôle s'imposent par exemple lorsqu'une validation complète sur la machine cible n'est pas réalisable ou que les objectifs en termes de performances sont difficiles à atteindre.

L'adaptation par rapport aux objectifs de qualité produit ou économiques s'effectue par le biais de la sélection d'exigences relatives à la qualité du produit, comme expliqué dans le paragraphe 7 du présent standard. Ce processus est fondé sur les objectifs de qualité du produit définis par le client.

5

Mise en œuvre du programme d'assurance produit logiciel

5.1 Organisation et responsabilités

5.1.1 Organisation

- a. Le fournisseur doit garantir qu'une structure organisationnelle est définie pour le développement logiciel et que des tâches et des responsabilités définies sont attribuées aux personnes impliquées.

5.1.2 Responsabilités et autorité

5.1.2.1

- a. Les responsabilités, l'autorité et les relations mutuelles du personnel chargé de la gestion, de l'exécution et de la vérification des travaux affectant la qualité du logiciel doivent être définies et documentées.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR].

5.1.2.2

- a. Les responsabilités et les interfaces de chaque organisation, externe ou interne, impliquée dans un projet doivent être définies et documentées.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR].

5.1.2.3

- a. La délégation des tâches d'assurance produit logiciel par un fournisseur à un fournisseur de niveau inférieur doit être effectuée de façon documentée et contrôlée, le fournisseur conservant ses responsabilités vis-à-vis du client.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR].

5.1.3 Ressources

5.1.3.1

- a. Le fournisseur doit fournir les ressources nécessaires à la réalisation des tâches d'assurance produit logiciel requises.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR].

5.1.3.2

- a. Les revues et audits réalisés sur les processus et les produits doivent l'être par du personnel qui ne soit pas directement impliqué dans les travaux exécutés.

5.1.4 Responsable/ingénieur assurance produit logiciel

5.1.4.1

- a. Le fournisseur doit identifier la personne responsable de l'assurance produit logiciel du projet (responsable/ingénieur AP logiciel).

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR].

5.1.4.2

- a. Le responsable/ingénieur assurance produit logiciel doit :
1. rendre des comptes au chef de projet (via le responsable assurance produit du projet, le cas échéant) ;
 2. disposer de l'autorité et de l'indépendance organisationnelles nécessaires pour proposer et maintenir un programme d'assurance produit logiciel conforme aux exigences d'assurance produit logiciel du projet ;
 3. avoir, autant que nécessaire, un accès direct vers les niveaux hiérarchiques supérieurs pour pouvoir s'acquitter des tâches qui lui incombent.

5.1.5 Formation

5.1.5.1

- a. Le fournisseur doit revoir les exigences du projet afin d'établir et de prendre les dispositions nécessaires pour permettre à l'équipe technique et de management d'acquérir ou de développer les ressources et les compétences requises.

RESULTAT ATTENDU : plan de formation [MGT, -; SRR].

5.1.5.2

- a. Le fournisseur doit conserver les enregistrements relatifs aux formations suivies.

RESULTAT ATTENDU: enregistrements relatifs à la formation et à l'expérience [PAF, -; -].

5.1.5.3

- a. Le fournisseur doit garantir que les profils de personnel formé à propos sont disponibles au moment voulu pour les activités et tâches planifiées.

5.1.5.4

- a. Le fournisseur doit déterminer les sujets de formation en fonction des outils, techniques, méthodologies et ressources informatiques spécifiques à utiliser dans le développement et la gestion du produit logiciel.

NOTE Le personnel peut suivre une formation afin d'acquérir des compétences et des connaissances spécifiques au domaine d'application du logiciel.

5.2 Gestion du programme d'assurance produit logiciel

5.2.1 Planification et contrôle de l'assurance produit logiciel

5.2.1.1

- a. Le fournisseur doit établir un plan assurance produit logiciel répondant aux exigences d'assurance produit logiciel en conformité avec les DRD de l'annexe B.
- b. Le plan assurance produit logiciel doit faire l'objet d'un document autonome ou d'une section du plan assurance produit global du fournisseur.

RESULTAT ATTENDU: plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

5.2.1.2

- a. Les manuels, standards ou procédures internes cités dans le plan assurance produit logiciel doivent faire partie intégrante du programme d'assurance produit logiciel du fournisseur.

5.2.1.3

- a. Le plan assurance produit logiciel doit être revu et mis à jour, si besoin est, à chaque jalon de sorte que les actions à entreprendre pour la phase suivante soient parfaitement définies.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; CDR, QR, AR, ORR].

5.2.1.4

- a. Avant la revue de recette, le fournisseur doit compléter le plan assurance produit logiciel en définissant les dispositions qualité relatives aux processus d'exploitation et de maintenance ou produire un plan assurance produit logiciel spécifique.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; AR].

5.2.1.5

- a. Le fournisseur doit joindre au plan assurance produit logiciel une matrice de conformité documentant la conformité à chaque exigence d'assurance produit logiciel applicable au projet ou à l'accord commercial.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

- b. Pour chaque exigence d'assurance produit logiciel, la matrice de conformité doit faire référence au document dans lequel le résultat attendu pour cette exigence est traité.

NOTE En ce qui concerne la conformité aux DRD, une déclaration générale de conformité est acceptée.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

5.2.2 Bilans qualité logiciel

5.2.2.1

- a. Le fournisseur doit régulièrement rendre compte de l'avancement de la mise en œuvre du programme d'assurance produit logiciel, le cas échéant dans le cadre du rapport d'assurance produit général du projet.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

5.2.2.2

- a. Le bilan qualité logiciel doit comprendre les éléments suivants :
1. une évaluation de la qualité actuelle du produit et des processus, s'appuyant sur les propriétés mesurées conformément aux systèmes de mesure définis dans le plan assurance produit logiciel ;
 2. les vérifications entreprises ;
 3. les problèmes détectés ;
 4. les problèmes résolus.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

5.2.2.3

- a. A chaque revue, le fournisseur doit livrer un bilan qualité logiciel de jalon traitant des activités d'assurance produit logiciel réalisées pendant les phases précédentes du projet.

RESULTAT ATTENDU : bilan qualité logiciel de jalon [PAF, SPAMR; SRR, PDR, CDR, QR, AR, ORR].

5.2.3 Audits

- a. Pour les audits logiciel, le paragraphe 5.2.3 de l'ECSS-Q-ST-10 doit être appliqué.

RESULTAT ATTENDU : plan et calendrier des audits [PAF, -; SRR].

5.2.4 Alertes

- a. Pour les alertes logiciel, le paragraphe 5.2.9 de l'ECSS-Q-ST-10 doit être appliqué.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. avis d'alerte préliminaire [PAF, -; -];
- b. avis d'alerte [PAF, -; -].

5.2.5 Problèmes logiciel

5.2.5.1

- a. Le fournisseur doit définir et mettre en œuvre des procédures pour l'enregistrement, l'analyse et la correction des problèmes logiciels rencontrés au cours du développement logiciel.

RESULTAT ATTENDU : procédures de traitement des problèmes logiciel [PAF, -; PDR].

5.2.5.2

- a. Un rapport de problème logiciel doit contenir les informations suivantes :
1. identification de l'élément logiciel ;
 2. description du problème ;
 3. solution recommandée ;
 4. décision finale ;
 5. modifications mises en œuvre (par exemple, documents, code et outils) ;
 6. essais exécutés une nouvelle fois.

RESULTAT ATTENDU : procédures de traitement des problèmes logiciel [PAF, -; PDR].

5.2.5.3

- a. Les procédures relatives aux problèmes logiciels doivent définir l'interface avec le système de traitement des non-conformités (c'est-à-dire les circonstances dans lesquelles un problème est qualifié de non-conformité).

RESULTAT ATTENDU : procédures de traitement des problèmes logiciel [PAF, -; PDR].

5.2.5.4

- a. Le fournisseur doit garantir la bonne application des procédures de traitement des problèmes logiciel.

5.2.6 Non-conformités

5.2.6.1

- a. Pour le traitement des non-conformités logiciel, l'ECSS-Q-ST-10-09 doit être appliquée.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *procédure de gestion des anomalies logiciel du plan assurance produit logiciel [PAF, SPAP; SRR];*
- b. *fiches d'anomalies [DJF, -; -].*
- b. La commission de traitement des non-conformités logiciel doit compter parmi ses membres au moins un représentant de l'organisme d'assurance produit logiciel et un représentant de l'organisme d'ingénierie logiciel.

RESULTAT ATTENDU : identification des experts logiciel dans la commission de traitement des anomalies logiciel [MGT, -; SRR].

5.2.6.2

- a. Le plan assurance produit logiciel doit indiquer le point du cycle de vie du logiciel à partir duquel s'appliquent les procédures relatives aux non-conformités.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

5.2.7 Exigences qualité et modèles qualité

5.2.7.1

- a. Des modèles qualité doivent être utilisés pour définir les exigences qualité logiciel.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR].

5.2.7.2

- a. Les caractéristiques suivantes doivent être utilisées pour définir le modèle qualité :
1. fonctionnalité ;
 2. fiabilité ;
 3. maintenabilité ;
 4. réutilisabilité ;
 5. sécurité ;
 6. sécurité ;
 7. facilité d'utilisation ;
 8. efficacité ;
 9. portabilité ;
 10. efficacité du développement logiciel.

NOTE 1 Les modèles qualité servent de base à l'identification des métriques sur les processus (voir le paragraphe 6.2.5) et les produits (voir le paragraphe 7.1.4).

NOTE 2 Les modèles qualité sont également traités dans la norme ISO/IEC 9126 ou le manuel ECSS-Q-HB-80-04.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR].

5.3 Management des risques et maîtrise des éléments critiques

5.3.1 Management des risques

- a. Le management des risques pour les logiciels doit être réalisé par référence croisée avec la politique de risque du projet, telle que définie dans l'ECSS-M-ST-80.

5.3.2 Maîtrise des éléments critiques

5.3.2.1

- a. Pour la maîtrise des éléments critiques, l'ECSS-Q-ST-10-04 doit être appliquée.

5.3.2.2

- a. Le fournisseur doit identifier les caractéristiques des éléments logiciel qui les qualifient pour les intégrer dans la liste des éléments critiques.

5.4 Sélection et suivi des fournisseurs

5.4.1 Sélection des fournisseurs

5.4.1.1

- a. Pour la sélection des fournisseurs, le paragraphe 5.4.1 de l'ECSS-Q-ST-20 doit être appliqué.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *résultats des audits et des évaluations avant contrat [PAF, -; -];*
- b. *enregistrements des sources d'approvisionnement [PAF, -; -].*

5.4.1.2

- a. Pour la sélection des fournisseurs de logiciels existants, incluant les logiciels contenus dans les équipements et unités sur étagère, les résultats attendus des paragraphes 6.2.7.2 à 6.2.7.6 doivent être mis à disposition.

RESULTAT ATTENDU : dossier des logiciels réutilisés [D]F, SRF; -].

5.4.2 Exigences pour les fournisseurs

5.4.2.1

- a. Le fournisseur doit établir des exigences d'assurance produit logiciel pour les fournisseurs de niveau immédiatement inférieur. Ces exigences doivent être adaptées au rôle joué par ces derniers dans le projet et inclure l'exigence de produire un plan assurance produit logiciel.

RESULTAT ATTENDU : exigences d'assurance produit logiciel applicables aux fournisseurs [PAF, -; SRR].

5.4.2.2

- a. Le fournisseur doit soumettre les exigences d'assurance produit logiciel applicables aux fournisseurs de niveau immédiatement inférieur à l'acceptation du client.

RESULTAT ATTENDU : exigences d'assurance produit logiciel applicables aux fournisseurs [PAF, -; SRR].

5.4.3 Surveillance des fournisseurs

5.4.3.1

- a. Le fournisseur doit surveiller le respect par les fournisseurs de niveau immédiatement inférieur des exigences d'assurance produit.

5.4.3.2

- a. Le processus de surveillance doit comprendre la revue et l'approbation des plans assurance produit des fournisseurs de niveau immédiatement inférieur, la vérification permanente des processus et des produits, et le suivi de la validation finale du produit.

5.4.3.3

- a. Le fournisseur doit s'assurer que des processus de développement logiciel sont définis et appliqués par les fournisseurs de niveau immédiatement inférieur, dans le respect des exigences d'assurance produit logiciel applicables aux fournisseurs.

RESULTAT ATTENDU : plan assurance produit logiciel des fournisseurs de niveau immédiatement inférieur [PAF, SPAP; PDR].

5.4.3.4

- a. Le fournisseur doit soumettre le plan assurance produit logiciel des fournisseurs de niveau immédiatement inférieur à l'acceptation du client.

RESULTAT ATTENDU : plan assurance produit logiciel des fournisseurs de niveau immédiatement inférieur [PAF, SPAP; PDR].

5.4.4 Classification de la criticité

- a. Le fournisseur doit donner aux fournisseurs de niveau inférieur les résultats des analyses de sécurité et de sûreté de fonctionnement réalisées à son niveau et aux niveaux supérieurs (voir les paragraphes 6.2.2.1 et 6.2.2.2), et notamment :

1. la classe de criticité des produits logiciels à développer ;
2. des informations sur les défaillances qui peuvent être provoquées à un niveau supérieur par les produits logiciels à développer.

RESULTAT ATTENDU : résultats des analyses de sécurité et de sûreté de fonctionnement des fournisseurs de niveau inférieur [RB, -; SRR].

5.5 Approvisionnement

5.5.1 Documents d'approvisionnement

- a. Pour les documents d'approvisionnement, le paragraphe 5.4.2 de l'ECSS-Q-ST-20 doit être appliqué.

5.5.2 Revue de la liste des composants logiciel acquis

- a. Le choix des logiciels acquis doit être décrit et soumis au client pour revue.

RESULTAT ATTENDU : plan de développement logiciel [MGT, SDP; SRR, PDR].

5.5.3 Caractérisation des approvisionnements

- a. Pour chacun des éléments logiciel, les données suivantes doivent être fournies :

1. critères de commande ;

NOTE Par exemple : versions, options et extensions.

2. critères de contrôle à la réception ;
3. solutions de secours en cas d'indisponibilité du produit ;
4. dispositions contractuelles convenues avec le fournisseur pour le développement, la maintenance et les mises à niveau de nouvelles versions.

RESULTAT ATTENDU : données d'approvisionnement [MGT, -; SRR, PDR].

5.5.4 Identification

- a. Tous les logiciels acquis doivent être identifiés et enregistrés par la gestion de configuration.

5.5.5 Contrôle

- a. Le fournisseur doit soumettre les logiciels acquis à un contrôle à la réception planifié, conforme au paragraphe 5.4.4 de l'ECSS-Q-ST-20 et aux critères de contrôle à la réception définis dans le paragraphe 5.5.3.

RESULTAT ATTENDU : compte rendu des contrôles à la réception [PAF, -; PDR, CDR, QR].

5.5.6 Export

- a. Les contraintes d'export doivent être identifiées.

5.6 Outils et environnement support

5.6.1 Méthodes et outils

5.6.1.1

- a. Les méthodes et outils à utiliser pour toutes les activités du cycle de développement (notamment l'analyse des exigences, la spécification des logiciels, la modélisation, la conception, le codage, la validation, les essais, la gestion de configuration, la vérification et l'assurance produit) doivent être identifiés par le fournisseur et acceptés par le client.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

5.6.1.2

- a. Le choix des méthodes et outils de développement doit être justifié au travers d'essais ou d'évaluations documentées démontrant que :

1. l'équipe de développement possède l'expérience ou la formation nécessaire pour les utiliser ;
2. ces méthodes et outils sont adaptés aux caractéristiques fonctionnelles et opérationnelles du produit ;
3. les outils sont disponibles (dans un environnement matériel approprié) pendant toute la durée du développement et de la maintenance du produit.

RESULTAT ATTENDU : bilan qualité logiciel de jalon [PAF, SPAMR; SRR, PDR].

5.6.1.3

- a. La bonne utilisation des méthodes et des outils doit être vérifiée et tracée.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

5.6.2 Sélection de l'environnement de développement

5.6.2.1

- a. L'environnement de développement logiciel doit être sélectionné en fonction des critères suivants :

1. disponibilité ;
2. compatibilité ;
3. performances ;
4. maintenance ;
5. pérennité et cohérence technique vis-à-vis des équipements opérationnels ;
6. évaluation du produit par rapport aux exigences, y compris la catégorie de criticité ;
7. disponibilité de la documentation support ;
8. conditions de recette et de garantie ;
9. conditions d'installation, de préparation, de formation et d'utilisation ;
10. conditions de maintenance, y compris les possibilités d'évolutions ;
11. contraintes de copyright et de droits de propriété intellectuelle ;
12. dépendance vis-à-vis d'un fournisseur particulier.

RESULTAT ATTENDU : plan de développement logiciel [MGT, SDP; SRR, PDR].

5.6.2.2

- a. L'adéquation de l'environnement de développement logiciel doit être justifiée.

RESULTAT ATTENDU : plan de développement logiciel [MGT, SDP; SRR, PDR].

5.6.2.3

- a. La disponibilité de l'environnement de développement logiciel pour les développeurs et autres utilisateurs doit être vérifiée avant le début de chaque phase de développement.

5.7 Processus d'évaluation et d'amélioration

5.7.1 Evaluation des processus

- a. Le fournisseur doit surveiller et contrôler l'efficacité des processus utilisés pendant le développement du logiciel, y compris les processus pertinents correspondants aux services demandés à d'autres entités organisationnelles extérieures à l'équipe projet.

NOTE L'évaluation et l'amélioration des processus réalisées au niveau de l'organisation peuvent servir de preuves de conformité pour le projet.

RESULTAT ATTENDU : enregistrements des évaluations des processus logiciel : plan du programme général d'évaluation et d'amélioration [PAF, -, -].

5.7.2 Processus d'évaluation

5.7.2.1

- a. Le modèle et la méthode d'évaluation des processus à utiliser lors de la réalisation d'une évaluation des processus logiciel doivent être documentés.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. enregistrement des évaluations des processus logiciel : modèle d'évaluation [PAF, -, -];
- b. enregistrement des évaluations des processus logiciel : méthode d'évaluation [PAF, -, -].

5.7.2.2

- a. Les évaluations réalisées et les modèles d'évaluation des processus utilisés doivent être conformes à la norme ISO/IEC 15504 (partie 2).

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. enregistrement des évaluations des processus logiciel : preuve de conformité au modèle d'évaluation des processus [PAF, -, -];
- b. enregistrement des évaluations des processus logiciel : méthode d'évaluation [PAF, -, -].

NOTE 1 Le modèle et la méthode documentés dans l'ECSS-Q-HB-80-02 sont conformes à la norme ISO/IEC 15504 (Partie 2).

NOTE 2 Le modèle CMMI actuel n'est pas totalement conforme à la norme ISO/IEC 15504. Il peut cependant être utilisé à condition que la méthode SCAMPI A soit appliquée.

5.7.2.3

- a. La conformité aux exigences du projet du modèle d'évaluation des processus, de la méthode, du domaine d'application de l'évaluation, des résultats et des évaluateurs doit être vérifiée.

NOTE 1 Voici deux exemples de domaines d'application de l'évaluation : unité organisationnelle évaluée et processus évalués.

NOTE 2 L'ECSS-Q-HB-80-02 fournit un modèle de référence de processus spécifiques au domaine spatial et leurs indicateurs.

RESULTAT ATTENDU : enregistrement des évaluations des processus logiciel : preuve de reconnaissance des évaluations des processus logiciel [PAF, -; -].

5.7.2.4

- a. Les évaluations, menées conformément à l'ECSS-Q-HB-80-02, doivent être réalisées par un évaluateur compétent tandis que l'équipe d'évaluation peut être constituée d'évaluateurs compétents ou d'évaluateurs désignés à titre provisoire.

NOTE 1 Pour les autres modèles d'évaluation conformes à la norme ISO/IEC 15504 (Partie 2) : les évaluateurs certifiés par l'INTRSA sont considérés comme des évaluateurs compétents.

NOTE 2 Lorsque la méthode CMMI/SCAMPI A est utilisée, les évaluateurs autorisés par le SEI sont considérés comme des évaluateurs compétents.

RESULTAT ATTENDU : enregistrement des évaluations des processus logiciel : justification d'évaluateur compétent [PAF, -; -].

5.7.3 Amélioration des processus

5.7.3.1

- a. Les résultats de l'évaluation doivent servir de retour d'expérience pour améliorer, le cas échéant, les processus réalisés, recommander des évolutions en direction du projet et déterminer les besoins en progrès technologiques.
- b. Les fournisseurs doivent garantir qu'ils utilisent les résultats des évaluations précédentes dans leurs activités projet.

RESULTAT ATTENDU : enregistrements des évaluations des processus logiciel : plan d'amélioration [PAF, -; -].

5.7.3.2

- a. L'amélioration des processus doit être réalisée selon un procédé d'amélioration des processus documenté.

NOTE 1 Pour obtenir la définition du procédé d'amélioration des processus, voir l'ECSS-Q-HB-80-02.

NOTE 2 Pour CMMI, l'amélioration des processus est décrite dans le processus OPF (Organizational Process Focus – orientation du processus organisationnel).

RESULTAT ATTENDU : enregistrements des évaluations des processus logiciel : processus d'amélioration [PAF, -; -].

5.7.3.3

- a. Des preuves de l'amélioration dans les processus mis en œuvre ou dans la documentation projet doivent être fournies.

NOTE Voir l'ECSS-Q-HB-80-02.

RESULTAT ATTENDU : enregistrements des évaluations des processus logiciel : preuve des améliorations [PAF, -; -].

6**Assurance des processus logiciel**

6.1 Cycle de vie du développement logiciel**6.1.1 Définition du cycle de vie**

- a. Le cycle de vie du développement logiciel doit être défini ou référencé dans le plan assurance produit logiciel.
- b. Les caractéristiques suivantes du cycle de vie logiciel doivent être définies :
 - 1. phases ;
 - 2. entrées et résultats de chaque phase ;
 - 3. état d'achèvement des résultats de phase ;
 - 4. jalons ;
 - 5. dépendances ;
 - 6. responsabilités ;
 - 7. rôle du client dans chaque revue de jalon, conformément aux standards ECSS-M-ST-10 et ECSS-M-ST-10-01.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

6.1.2 Objectifs qualité des processus

- a. Dans la définition du cycle de vie, des jalons et documents associés, les objectifs qualité doivent être utilisés.

6.1.3 Revue de la définition du cycle de vie

- a. Le cycle de vie logiciel doit être revu par rapport aux exigences contractuelles relatives à l'ingénierie et à l'assurance produit logiciel.

6.1.4 Ressources du cycle de vie

- a. Le cycle de vie logiciel doit être revu afin de vérifier l'adéquation et la disponibilité des ressources nécessaires à sa mise en œuvre par toutes les fonctions impliquées dans son déroulement.

6.1.5 Planification du processus de validation logiciel

- a. Un jalon (SW TRR, conformément au paragraphe 5.3.5.1 de l'ECSS-E-ST-40) doit être planifié immédiatement avant le début du processus de validation logiciel afin de vérifier que :
 - 1. l'état du logiciel est compatible avec le début des activités de validation ;
 - 2. les ressources, plans assurance produit logiciel, documents de test et de validation, simulateurs et autres moyens techniques nécessaires sont disponibles et prêts à l'emploi.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

6.2 Exigences applicables à tous les processus d'ingénierie logiciel

6.2.1 Documentation des processus

6.2.1.1

- a. Les activités suivantes doivent être couvertes par des plans propres aux logiciels ou des plans de projet généraux :
 - 1. développement ;
 - 2. spécification, conception et documents à produire pour le client ;
 - 3. gestion de configuration et de documentation ;
 - 4. activités de vérification, de test et de validation ;
 - 5. maintenance.

RESULTAT ATTENDU : plans de projet logiciel [MGT, MF, DJF].

6.2.1.2

- a. Tous les plans doivent être terminés avant le début des activités associées.

RESULTAT ATTENDU : plans de projet logiciel [MGT, MF, DJF].

6.2.1.3

- a. Tous les plans doivent être mis à jour à chaque jalon afin de refléter toutes les modifications effectuées au cours du développement.

RESULTAT ATTENDU : plans de projet logiciel [MGT, MF, DJF].

6.2.1.4

- a. Le plan assurance produit logiciel doit identifier tous les plans à produire et à utiliser, les relations entre eux et les délais associés à leur préparation et à leur mise à jour.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

6.2.1.5

- a. Chaque plan doit être revu par rapport aux exigences contractuelles correspondantes.

6.2.1.6

- a. Les procédures et les standards du projet doivent traiter tous les types de produits logiciel inclus dans le projet.

RESULTAT ATTENDU : procédures et standards [PAF, -; PDR].

6.2.1.7

- a. Toutes les procédures et les standards du projet doivent être terminés avant le début des activités associées.

RESULTAT ATTENDU : procédures et standards [PAF, -; PDR].

6.2.1.8

- a. Chaque procédure ou standard doit être revu par rapport aux plans et aux exigences contractuelles correspondants.

6.2.1.9

- a. Avant le début de toute activité, chaque procédure ou standard pour cette activité doit être revu par toutes les fonctions impliquées dans son application, en termes d'adéquation et de disponibilité des ressources intervenant dans sa mise en œuvre.

6.2.2 Sûreté de fonctionnement des logiciels et sécurité

6.2.2.1

- a. En ce qui concerne les analyses de niveau système menant à la classification de la criticité des produits logiciel en fonction de la gravité des conséquences des défaillances, le Tableau 6-1 de l'ECSS-Q-ST-40 et le Tableau 5-1 de l'ECSS-Q-ST-30 doivent être appliqués.

RESULTAT ATTENDU : classification de la criticité des produits logiciel [PAF, -; SRR, PDR].

6.2.2.2

- a. Le fournisseur doit soumettre les produits logiciel à une analyse de sûreté de fonctionnement et de sécurité de niveau logiciel, conformément

aux exigences des standards ECSS-Q-ST-30 et ECSS-Q-ST-40 et à l'aide des résultats des analyses de sûreté de fonctionnement et de sécurité de niveau système, afin de déterminer la criticité de chaque composant logiciel.

RESULTAT ATTENDU : rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels [PAF, -; PDR].

6.2.2.3

- a. Le fournisseur doit identifier les méthodes et les techniques d'analyse de sûreté de fonctionnement et de sécurité de niveau logiciel à utiliser aux niveaux de la spécification technique et de la conception.
- b. Les méthodes et les techniques d'analyse de sûreté de fonctionnement et de sécurité des logiciels doivent être acceptées par le fournisseur et le client.

NOTE L'ECSS-Q-HB-80-03 donne des informations sur les méthodes et les techniques qui peuvent être appliquées. Par exemple :

- analyse des modes de défaillances, de leurs effets et de leur criticité de niveau logiciel (pour mener cette analyse, voir également l'ECSS-Q-ST-30-02) ;
- analyse de l'arbre de panne des logiciels ;
- analyse des causes communes de défaillances des logiciels.

RESULTAT ATTENDU : classification de la criticité des composants logiciel [PAF, -; PDR].

6.2.2.4

- a. Sur la base des résultats de l'analyse de criticité logiciel, le fournisseur doit appliquer des mesures d'ingénierie destinées à réduire le nombre de composants logiciel critiques, ainsi que les risques associés (voir le paragraphe 6.2.3).

6.2.2.5

- a. Le fournisseur doit rendre compte de l'avancement de la mise en œuvre et de la vérification des recommandations issues de l'analyse de sûreté de fonctionnement et de sécurité de niveau logiciel.

RESULTAT ATTENDU : rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels [PAF, -; CDR, QR, AR].

6.2.2.6

- a. Le fournisseur doit mettre à jour l'analyse de sûreté de fonctionnement et de sécurité des logiciels à chaque jalon du développement logiciel afin de confirmer la catégorie de criticité des composants logiciel.

RESULTAT ATTENDU : rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels [PAF, -; CDR, QR, AR].

6.2.2.7

- a. Le fournisseur doit communiquer les résultats de l'analyse de sûreté de fonctionnement et de sécurité des logiciels afin qu'ils soient intégrés dans les analyses de sûreté de fonctionnement et de sécurité de niveau système. Les points suivants doivent notamment être traités :
1. modes de défaillances supplémentaires identifiés au niveau de la conception logiciel ;
 2. recommandations relatives aux activités de niveau système.

NOTE Par exemple : introduction d'inhibitions matériel et modifications de l'architecture système.

RESULTAT ATTENDU : rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels [PAF, -; PDR, CDR].

6.2.2.8

- a. Dans le cadre des activités d'analyse des exigences logiciel (voir le paragraphe 6.3.2), le fournisseur doit contribuer à l'analyse des interactions matériel logiciel (HSIA) en identifiant, pour chaque défaillance du matériel incluse dans la HSIA, les exigences définissant le comportement du logiciel en cas d'apparition de cette défaillance du matériel.

6.2.2.9

- a. Pendant la vérification et la validation des exigences logiciel issues de l'analyse des interactions matériel logiciel, le fournisseur doit vérifier que le logiciel réagit correctement aux défaillances du matériel et qu'aucun défaut de fonctionnement logiciel indésirable qui pourrait conduire à une défaillance système, ne se produit.

6.2.3 Traitement des logiciels critiques

6.2.3.1

- a. Le fournisseur doit définir et mettre en œuvre des mesures permettant d'éviter la propagation des défaillances entre des composants logiciel appartenant à des catégories de criticité différentes.
- b. Ceci peut être obtenu au moyen de mesures de conception telles que l'utilisation de plates-formes matérielles séparées, l'isolation des processus logiciel ou l'interdiction du partage de la mémoire (ségrégation et partitionnement).

Les composants logiciel dont la défaillance peut entraîner des défaillances sur des composants de criticité supérieure doivent être classés en fonction des conséquences de ces défaillances.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *plan assurance produit logiciel [PAF, SPAP; PDR, CDR];*
- b. *rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels [PAF, -; PDR, CDR, QR, AR].*

6.2.3.2

- a. Le fournisseur doit définir, justifier et appliquer des mesures pour assurer la sûreté de fonctionnement et la sécurité des logiciels critiques.

NOTE Ces mesures peuvent comprendre :

- l'utilisation d'une conception logiciel ou de méthodes ayant donné de bons résultats dans une application similaire ;
- l'insertion de fonctionnalités pour l'isolation et de traitement des défaillances (voir l'ECSS-Q-HB-80-03, analyse des modes de défaillances logiciel, de leurs effets et de leur criticité) ;
- des techniques de programmation défensives, telles que la vérification des entrées et les contrôles de la cohérence ;
- l'utilisation d'un « sous-ensemble sûr » de langage de programmation ;
- l'utilisation d'un langage de conception formel pour obtenir des preuves formelles ;
- une couverture des branches du code à 100 % au niveau des tests unitaires ;
- le contrôle complet du code source ;
- des essais avec témoins ou menés par une équipe indépendante ;
- la collecte et l'analyse des statistiques sur les défaillances ;
- la suppression du code désactivé ou la démonstration par une combinaison d'analyse et d'essais que les moyens par lesquels un tel code peut être exécuté par inadvertance sont empêchés, isolés ou éliminés.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR, CDR].

6.2.3.3

- a. L'application des mesures choisies pour le traitement des logiciels critiques doit être vérifiée.

RESULTAT ATTENDU : bilan qualité logiciel de jalon [PAF, SPAMR; PDR, CDR, QR, AR].

6.2.3.4

- a. Les logiciels critiques doivent faire l'objet d'essais de non régression après :

1. toute évolution de fonctionnalité de la plate-forme matérielle sous jacente ;

NOTE Par exemple : jeu d'instructions d'un processeur.

2. toute évolution des outils ayant une incidence directe ou indirecte sur la génération du code exécutable.

NOTE Dans le cas de modifications mineures sur les outils ayant une incidence sur la génération du code exécutable, une comparaison binaire du code exécutable généré par les différents outils peut être utilisée pour vérifier qu'aucun changement n'a été introduit.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR, CDR].

6.2.3.5

- a. La nécessité de soumettre les logiciels critiques à une vérification et à une validation complémentaires doit être analysée après :

1. toute évolution de fonctionnalité ou de performance de la plate-forme matérielle sous jacente ;
2. toute évolution de l'environnement dans lequel fonctionnent les logiciels ou la plate-forme matérielle.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR, CDR].

6.2.3.6

- a. Le code mort identifié doit être supprimé et la nécessité de procéder à une nouvelle vérification et à une nouvelle validation doit être analysée.

6.2.3.7

- a. Les tests unitaires et les essais d'intégration doivent être (ré)exécutés sur le code non instrumenté.

6.2.3.8

- a. Les essais de validation doivent être (ré)exécutés sur le code non instrumenté.

6.2.4 Gestion de configuration des logiciels

6.2.4.1

- a. L'ECSS-M-ST-40 doit être appliquée pour la gestion de configuration des logiciels avec les exigences complémentaires suivantes.

6.2.4.2

- a. Le système de gestion de configuration des logiciels doit permettre de générer à nouveau toute version de référence à partir de sauvegardes.

RESULTAT ATTENDU : plan de gestion de configuration des logiciels [MGT, SCMP; SRR, PDR].

6.2.4.3

- a. Le dossier descriptif de configuration et le document de version du logiciel doivent être fournis à chaque livraison du logiciel.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. dossier descriptif de configuration du logiciel [DDF, SCF; -];
- b. document de version du logiciel [DDF, SRelD;-].

6.2.4.4

- a. Le dossier descriptif de configuration du logiciel doit être disponible et à jour pour chaque jalon du projet.

RESULTAT ATTENDU : dossier descriptif de configuration du logiciel [DDF, SCF; CDR, QR, AR, ORR].

6.2.4.5

- a. Tous les composants de l'outil de génération de code qui peuvent être configurés par l'utilisateur doivent être gérés en configuration.
- b. Les procédures de maîtrise des modifications définies pour le projet doivent prendre en compte les aspects particuliers de ces composants.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. dossier descriptif de configuration du logiciel [DDF, SCF; CDR, QR, AR, ORR] ;
- b. plan de gestion de configuration des logiciels [MGT, SCMP; SRR, PDR].

6.2.4.6

- a. Le fournisseur doit garantir que toutes les modifications autorisées sont mises en œuvre conformément au plan de gestion de configuration des logiciels.

RESULTAT ATTENDU : Modifications autorisées - Dossier descriptif de configuration du logiciel [DDF, SCF; CDR, QR, AR, ORR].

6.2.4.7

- a. Les documents suivants doivent être contrôlés (voir le paragraphe 5.2.5 de l'ECSS-Q-ST-10) :

1. documents procéduraux décrivant le système qualité qui doit être appliqué durant le cycle de vie du logiciel ;
2. documents de planification décrivant la planification et l'avancement des activités ;
3. documents décrivant un produit logiciel particulier, comprenant :
 - (a) les entrées de phase de développement ;
 - (b) les résultats de phase de développement ;
 - (c) les plans de vérification et de validation et les résultats ;
 - (d) les spécifications des cas de test, les procédures et les rapports d'essai ;
 - (e) les matrices de traçabilité ;
 - (f) la documentation destinée aux opérateurs et utilisateurs des logiciels et du système ;
 - (g) la documentation de maintenance.

6.2.4.8

- a. Le fournisseur doit identifier une méthode et un outil pour protéger les logiciels produits contre la corruption.

NOTE Par exemple : source, exécutable, données.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *plan assurance produit logiciel [PAF, SPAP; SRR, PDR] ;*
- b. *dossier descriptif de configuration du logiciel [DDF, SCF; CDR, QR, AR, ORR].*

6.2.4.9

- a. Le fournisseur doit définir un calcul de type checksum pour les logiciels opérationnels livrés.

NOTE Par exemple : binaire exécutable, base de données.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

6.2.4.10

- a. La valeur du checksum doit être fournie dans le dossier descriptif de configuration du logiciel, à chaque livraison de logiciel.

RESULTAT ATTENDU : dossier descriptif de configuration du logiciel [DDF, SCF; -].

6.2.4.11

- a. Les supports utilisés pour la livraison du logiciel au client doivent être marqués par le fournisseur en indiquant au minimum les informations suivantes :

1. le nom du logiciel ;
2. le numéro de version ;
3. la référence au dossier descriptif de configuration du logiciel.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. plan assurance produit logiciel [PAF, SPAP; SRR, PDR] ;*
- b. marquages [DDF, -; -].*

6.2.5 Métriques sur les processus

6.2.5.1

- a. Des métriques doivent être utilisées pour gérer le développement et évaluer la qualité des processus de développement.

NOTE Les métriques sur les processus s'appuient sur les modèles qualité (voir le paragraphe 5.2.7).

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

6.2.5.2

- a. Les métriques sur les processus doivent être régulièrement collectées, conservées et analysées en appliquant des modèles et des procédures qualité.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

6.2.5.3

- a. Les métriques sur les processus de base suivantes doivent être utilisées au sein de l'organisation du fournisseur :

1. durée : durée d'achèvement des phases/tâches par rapport au planning prévu ;
2. effort : effort consacré aux diverses phases/tâches par rapport aux prévisions.

RESULTAT ATTENDU : rapport sur les métriques internes.

6.2.5.4

- a. Des métriques sur les processus doivent être utilisées au sein de l'organisation du fournisseur et transmises au client, incluant :
1. nombre de problèmes détectés au cours des vérifications ;
 2. nombre de problèmes détectés lors des essais d'intégration et de validation et lors de l'utilisation.

NOTE Voir également la partie gestion des problèmes logiciel décrite dans le paragraphe 5.2.5.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.2.5.5

- a. Les rapports sur les métriques doivent être inclus dans les bilans qualité logiciel.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.2.6 Vérification

6.2.6.1

- a. Les activités de vérification des exigences qualité doivent être spécifiées dans la définition du plan de vérification.

NOTE La vérification comprend différentes techniques telles que les revues, les contrôles, les essais, les contrôles transverses, les relectures croisées, les contrôles de programmation, les simulations sur modèle et de nombreux types d'analyses telles que l'analyse de traçabilité, la preuve formelle ou l'analyse d'arbre de panne.

RESULTAT ATTENDU : plan de vérification logiciel [DJF, SVerP; PDR].

6.2.6.2

- a. La conformité des résultats de chaque activité de développement doit être vérifiée par rapport à des critères prédéfinis .
- b. Seuls les résultats ayant fait l'objet de vérifications planifiées doivent être utilisés comme entrées pour les activités suivantes.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.2.6.3

- a. Un récapitulatif des activités d'assurance associées au processus de vérification et de ses résultats doit être inséré dans les bilans qualité logiciel.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.2.6.4

- a. La réalisation des actions associées aux problèmes logiciel rapportés au cours de la vérification doit être vérifiée et enregistrée.

RESULTAT ATTENDU : fiches d'anomalies logicielles [DJF, -; SRR, PDR, CDR, QR, AR, ORR].

6.2.6.5

- a. Les logiciels contenant du code désactivé doivent être vérifiés spécifiquement pour garantir que le code désactivé ne peut pas être activé ou que son activation accidentelle ne peut pas nuire au fonctionnement du système.

RESULTAT ATTENDU : rapport de vérification logiciel [DJF, SVR; CDR, QR, AR].

6.2.6.6

- a. Les logiciels contenant du code configurable doivent être vérifiés spécifiquement pour garantir qu'une configuration non désirée ne peut pas être activée lors de l'exécution du logiciel ni être prise en compte lors de la génération du code.

RESULTAT ATTENDU : rapport de vérification logiciel [DJF, SVR; CDR, QR, AR].

6.2.6.7

- a. Le fournisseur doit s'assurer que :
1. les activités de vérification planifiées sont suffisantes pour garantir que les produits de chaque phase sont conformes aux exigences applicables ;
 2. les activités de vérification sont effectuées conformément au plan.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.2.6.8

- a. Les revues et les contrôles doivent être menés selon des critères définis et selon le niveau d'indépendance défini entre le responsable de la revue et l'auteur de l'élément passé en revue.

6.2.6.9

- a. Chaque revue et chaque contrôle doivent s'appuyer sur un plan écrit ou une procédure écrite.

NOTE Pour les revues de projet, le paragraphe 5.3.3.3, point b et l'Annexe P de l'ECSS-E-ST-40 s'appliquent.

RESULTAT ATTENDU : plans ou procédures de revue et de contrôle [PAF, -; -].

6.2.6.10

- a. Les plans ou procédures de revue/contrôle doivent préciser :
1. les éléments revus ou contrôlés ;
 2. le responsable de l'action ;
 3. les participants ;
 4. les moyens de revue/contrôle (par exemple, outils ou fiche de vérification) ;
 5. la nature du compte rendu.

RESULTAT ATTENDU : plans ou procédures de revue et de contrôle [PAF, -; -].

6.2.6.11

- a. Les rapports de revue et de contrôle doivent :
1. faire référence à la procédure ou au plan de revue/contrôle correspondant ;
 2. identifier l'élément passé en revue, l'auteur, le responsable de la revue, les critères de revue et les résultats de la revue.

RESULTAT ATTENDU : rapports de revue et de contrôle [PAF, -; -].

6.2.6.12

- a. Les matrices de traçabilité (telles que définies dans le paragraphe 5.8 de l'ECSS-E-ST-40) doivent être vérifiées à chaque jalon.

RESULTAT ATTENDU : bilan qualité logiciel de jalon [PAF, SPAMR; SRR, PDR, CDR, QR, AR, ORR].

6.2.6.13

- a. Une vérification indépendante des logiciels doit être réalisée par un tiers.
- b. La vérification indépendante des logiciels doit comporter différentes revues, contrôles, analyses, simulations, essais et audits.

NOTE Cette exigence n'est applicable que lorsque les risques associés au projet justifient les coûts induits. Le client peut envisager un niveau d'indépendance moindre, par exemple une équipe indépendante au sein de la même entité.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. plan ISVV [DJF, -; SRR, PDR] ;
- b. rapport ISVV [DJF, -; PDR, CDR, QR, AR].

6.2.7 Réutilisation de logiciels existants

6.2.7.1 Généralités

Les exigences du paragraphe 6.2.7 ne s'appliquent pas aux outils et à l'environnement de développement logiciel soumis aux exigences du paragraphe 5.6.

6.2.7.2

- a. Les avantages que représente l'utilisation de logiciels existants par rapport à un nouveau développement doivent être analysés.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *approche de réutilisation des logiciels, notamment l'approche de delta qualification [PAF, SPAP; SRR, PDR] ;*
- b. *dossier des logiciels réutilisés [DJF, SRF; SRR, PDR].*

6.2.7.3

- a. Le logiciel existant doit être évalué par rapport aux exigences fonctionnelles, de performances et de qualité applicables.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *approche de réutilisation des logiciels, notamment l'approche de delta qualification [PAF, SPAP; SRR, PDR] ;*
- b. *dossier des logiciels réutilisés [DJF, SRF; SRR, PDR].*

6.2.7.4

- a. Le niveau de qualité du logiciel existant doit être analysé en fonction des exigences du projet, de la criticité des fonctions du système mises en œuvre, en prenant en compte les aspects suivants :

1. documentation des exigences logiciel ;
2. documentation de l'architecture et de la conception détaillée du logiciel ;
3. traçabilité ascendante et descendante entre les exigences système, les exigences logiciel, la conception et le code ;
4. documentation et couverture des tests unitaires ;
5. documentation et couverture des essais d'intégration ;
6. documentation et couverture de la validation ;
7. rapports de vérification ;
8. performances ;

NOTE Par exemple : occupation mémoire, charge de l'UC.

9. performances en exploitation ;
10. non-conformités et dérogations résiduelles ;
11. documentation utilisateur pour l'exploitation;
- NOTE Par exemple : manuel utilisateur.
12. qualité du code (respect des standards de codage, des métriques).

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *approche de réutilisation des logiciels, notamment l'approche de delta qualification [PAF, SPAP; SRR, PDR] ;*

- b. dossier des logiciels réutilisés [DJF, SRF; SRR, PDR].

6.2.7.5

- a. Les résultats de l'analyse des logiciels réutilisés doivent être conservés dans le dossier des logiciels réutilisés, accompagnés d'une évaluation du niveau possible de réutilisation et d'une description des hypothèses et des méthodes appliquées lors de l'estimation du niveau de réutilisation.

NOTE Voici quelques exemples de résultats de l'analyse des logiciels réutilisés : références détaillées aux documents d'exigences et de conception, rapports d'essai et résultats de couverture.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. approche de réutilisation de logiciels, notamment l'approche de delta qualification [PAF, SPAP; SRR, PDR] ;
- b. dossier des logiciels réutilisés [DJF, SRF; SRR, PDR].

6.2.7.6

- a. L'analyse de la réutilisabilité des logiciels existants doit être complétée par une évaluation des aspects suivants :
1. conditions de recette et de garantie ;
 2. disponibilité de la documentation de soutien ;
 3. conditions d'installation, de préparation, de formation et d'utilisation ;
 4. identification et enregistrement par la gestion de configuration ;
 5. responsabilités et conditions de maintenance, y compris les possibilités d'évolution ;
 6. pérennité et validité des méthodes et outils utilisés dans le développement initial et dont la réutilisation est envisagée ;
 7. contraintes de copyright et de droits de propriété intellectuelle (droits de modification) ;
 8. conditions de licence ;
 9. contraintes d'export.

RESULTAT ATTENDU : dossier des logiciels réutilisés [DJF, SRF; SRR, PDR].

6.2.7.7

- a. Des actions correctives doivent être identifiées, documentées dans le dossier des logiciels réutilisés et appliquées aux logiciels réutilisés ne satisfaisant pas les exigences applicables relatives aux aspects évoqués dans les paragraphes 6.2.7.2 à 6.2.7.6.

RESULTAT ATTENDU : dossier des logiciels réutilisés [DJF, SRF ; SRR, PDR].

6.2.7.8

- a. Des techniques de rétro ingénierie doivent être appliquées pour créer la documentation manquante et répondre aux exigences de couverture de vérification et de validation.
- b. Pour les produits logiciel pour lesquels on ne dispose pas de données de cycle de vie issues de développements antérieurs et pour lesquels les techniques de rétro ingénierie ne sont pas totalement applicables, les méthodes suivantes doivent être appliquées :
 1. génération de documents de validation et de vérification s'appuyant sur la documentation utilisateur disponible (par exemple, manuel utilisateur) et exécution de tests en vue d'atteindre le niveau de couverture requis ;
 2. utilisation de l'historique du produit en service pour justifier l'adéquation du produit par rapport à l'application actuelle, cet historique comprenant des informations relatives à :
 - (a) la pertinence de l'historique du produit en service pour le nouvel environnement d'exploitation ;
 - (b) la gestion de configuration et la maîtrise des modifications du produit logiciel ;
 - (c) l'efficacité du suivi des problèmes ;
 - (d) les taux d'erreurs réels et les enregistrements de maintenance ;
 - (e) l'impact des modifications.

RESULTAT ATTENDU : dossier des logiciels réutilisés [DJF, SRF ; SRR, PDR].

6.2.7.9

- a. Le dossier des logiciels réutilisés doit être mis à jour au niveau des jalons du projet afin de refléter les résultats des actions correctives identifiées pour les logiciels réutilisés ne satisfaisant pas les exigences du projet.

RESULTAT ATTENDU : dossier des logiciels réutilisés [DJF, SRF ; CDR, QR, AR].

6.2.7.10

- a. Tous les logiciels réutilisés doivent être gérés en configuration.

6.2.7.11

- a. L'état de configuration détaillé des logiciels réutilisés de référence doit être fourni au client dans le dossier des logiciels réutilisés pour acceptation.

RESULTAT ATTENDU : dossier des logiciels réutilisés [DJF, SRF ; SRR, PDR, CDR, QR, AR].

6.2.8 Génération automatique de codes

6.2.8.1

- a. Lors de la sélection des outils de génération automatique de codes, le fournisseur doit évaluer les aspects suivants :
1. évolution des outils par rapport aux outils utilisant le code généré en entrée ;

NOTE Par exemple : compilateurs ou systèmes de gestion du code.
 2. adaptation des outils dans un souci de respect des standards du projet ;
 3. exigences de portabilité du code généré ;
 4. collecte des métriques exigées sur la conception et le code ;
 5. vérification des composants logiciel contenant le code généré ;
 6. gestion en configuration des outils, y compris des paramètres d'adaptation ;
 7. conformité aux standards ouverts ou libres.

6.2.8.2

- a. Les exigences relatives aux tests applicables au code généré automatiquement doivent garantir la réalisation des mêmes objectifs que ceux associés au code généré manuellement.

RESULTAT ATTENDU : documentation de validation et de test [DJF, SValP; PDR], [DJF, SVS; CDR, QR, AR], [DJF, SUIP; PDR, CDR].

6.2.8.3

- a. Le niveau de vérification et de validation requis pour l'outil de génération automatique doit être au moins équivalent à celui exigé pour le code généré si l'outil est utilisé pour s'affranchir de certaines activités de vérification ou de test sur le code cible.

6.2.8.4

- a. Des standards de modélisation doivent être définis et appliqués pour les outils de génération automatique de codes.

RESULTAT ATTENDU : standards de modélisation [PAF, -; SRR, PDR].

6.2.8.5

- a. Le respect des standards de modélisation doit être vérifié.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.2.8.6

- a. Le paragraphe 6.3.4 doit être appliqué au code généré automatiquement, à moins que le fournisseur ne démontre que ce code n'a pas besoin d'être modifié manuellement.

6.2.8.7

- a. La documentation de vérification et de validation doit distinguer les activités à réaliser pour le code généré manuellement des activités à réaliser pour le code généré de manière automatique.

RESULTAT ATTENDU : documentation de validation et des essais [DJF, SValP; RDP], [DJF, SVS; RCD, RQ, RA], [DJF, SUITP; RDP, RCD].

6.3 Exigences applicables à chacun des processus ou activités d'ingénierie logiciel

6.3.1 Traitement des exigences système relatives au logiciel

6.3.1.1

- a. Pour la définition des exigences système relatives au logiciel à spécifier dans les exigences de besoin, le paragraphe 5.2 de l'ECSS-E-ST-40 doit être appliqué.

6.3.1.2

- a. Les exigences de besoin doivent être soumises au contrôle de la documentation et à la gestion de configuration en tant qu'éléments de la documentation de développement.

6.3.1.3

- a. Pour la définition des exigences de besoin, les résultats de toutes les analyses de sécurité et de sûreté de fonctionnement (y compris ceux de la HSIA décrite dans le paragraphe 6.4.2.3 de l'ECSS-Q-ST-30) doivent être utilisés.

6.3.2 Analyse des exigences logiciel

6.3.2.1

- a. Les exigences de besoin doivent être analysées afin que les exigences logiciel puissent être définies de manière exhaustive et univoque dans la spécification technique.

6.3.2.2

- a. La spécification technique doit être soumise au contrôle de la documentation et à la gestion de configuration en tant qu'élément de la documentation de développement.

6.3.2.3

- a. Pour la définition de la spécification technique, les résultats de toutes les analyses de sécurité et de sûreté de fonctionnement (y compris ceux de la HSIA décrite dans le paragraphe 6.4.2.3 de l'ECSS-Q-ST-30) doivent être utilisés.

6.3.2.4

- a. En complément des exigences fonctionnelles, la spécification technique doit contenir toutes les exigences non fonctionnelles nécessaires à la satisfaction des exigences de base, comprenant au moins :
1. les performances ;
 2. la sécurité ;
 3. la fiabilité ;
 4. la robustesse ;
 5. la qualité ;
 6. la maintenabilité ;
 7. la gestion de configuration ;
 8. la sécurité informatique;
 9. la confidentialité ;
 10. les métriques ;
 11. la vérification et la validation.

NOTE Les exigences de performances comprennent les exigences de précision numérique.

RESULTAT ATTENDU : spécification des exigences logiciel [TS, SRS; PDR].

6.3.2.5

- a. Préalablement à l'élaboration de la spécification technique, le client et le fournisseur doivent au minimum se mettre d'accord sur les principes et règles suivants :
1. désignation des personnes (au sein des deux parties) responsables de l'établissement de la spécification technique ;
 2. méthodes d'acceptation des exigences et d'approbation des modifications ;
 3. efforts mis en œuvre pour éviter les mauvaises compréhensions, tels que la définition des termes et l'explication du fondement des exigences ;
 4. enregistrement et revue des résultats des discussions, au sein des deux parties.

6.3.3 Conception de l'architecture du logiciel et des éléments logiciel

6.3.3.1

- a. Le dossier de définition de la conception doit être soumis au contrôle de la documentation et à la gestion de configuration.

6.3.3.2

- a. Des standards de conception obligatoires et recommandées doivent être définis et appliqués.

RESULTAT ATTENDU : standards de conception [PAF, -, SRR, PDR].

6.3.3.3

- a. Pour les logiciels dans lesquels la précision numérique conditionne la réussite de la mission, des règles particulières de conception et de codage doivent être définies afin de garantir que le niveau de précision spécifié est atteint.

NOTE Par exemple : pour un sous-système de contrôle d'orbite et d'attitude, des composants de génération de données scientifiques.

RESULTAT ATTENDU : *plan assurance produit logiciel* [PAF, SPAP; PDR].

6.3.3.4

- a. Le respect des standards de conception doit être vérifié.

RESULTAT ATTENDU : *bilans qualité logiciel* [PAF, -, -].

6.3.3.5

- a. Le fournisseur doit définir des moyens, des critères et des outils afin de garantir que la complexité et la modularité de la conception répondent aux exigences qualité.
- b. L'évaluation de la conception doit être réalisée parallèlement au processus de conception afin que les retours puissent être transmis à l'équipe de conception logiciel.

RESULTAT ATTENDU : *plan assurance produit logiciel* [PAF, SPAP; PDR].

6.3.3.6

- a. La synthèse des résultats obtenus lors de l'évaluation de la complexité et de la modularité du logiciel et des actions correctives mises en œuvre doit être décrite dans les bilans qualité logiciel.

RESULTAT ATTENDU : *bilans qualité logiciel* [PAF, -, -].

6.3.3.7

- a. Le fournisseur doit passer en revue la documentation de conception pour garantir qu'elle contient le niveau d'information approprié pour les activités de maintenance.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. plan assurance produit logiciel [PAF, SPAP; PDR] ;
- b. bilans qualité logiciel [PAF, -; -].

6.3.4 Codage

6.3.4.1

- a. Des standards de codage (comprenant des conventions de nommage cohérentes et des règles sur les commentaires) doivent être spécifiés et appliqués.

RESULTAT ATTENDU : standards de codage [PAF, -; PDR].

6.3.4.2

- a. Les standards doivent être cohérents par rapport aux exigences qualité produit.

NOTE Les standards de codage dépendent des objectifs de qualité logiciel (voir le paragraphe 5.2.7).

RESULTAT ATTENDU : standards de codage [PAF, -; PDR].

6.3.4.3

- a. Les outils à utiliser dans la mise en œuvre et le contrôle de la conformité aux standards de codage doivent être identifiés dans le plan assurance produit avant le début des activités de codage.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR].

6.3.4.4

- a. Les standards de codage doivent être revus avec le client pour garantir qu'ils reflètent les exigences qualité produit.

RESULTAT ATTENDU : standards de codage et description des outils [PAF, -; PDR].

6.3.4.5

- a. L'utilisation de langages de programmation de bas niveau doit être justifiée.

RESULTAT ATTENDU : plan de développement logiciel [MGT, SDP; PDR].

6.3.4.6

- a. Le fournisseur doit définir des mesures, des critères et des outils afin de garantir que le code répond aux exigences qualité.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR].

- b. L'évaluation du code doit être réalisée parallèlement au processus de codage afin que les retours puissent être transmis aux programmeurs du logiciel.

6.3.4.7

- a. La synthèse des résultats de l'analyse du code et des actions correctives mises en œuvre doit être décrite dans les bilans qualité logiciel.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -, -].

6.3.4.8

- a. Le code doit être géré en configuration immédiatement après la réussite des tests unitaires.

6.3.5 Tests et validation

6.3.5.1

- a. Les tests doivent être réalisés conformément à une stratégie pour chaque niveau de test (c'est-à-dire, tests unitaires, intégration, validation des spécifications techniques, validation par rapport aux exigences de besoin, recette) comprenant :

1. les types des tests à effectuer ;

NOTE Par exemple : fonctionnel, aux limites, de performances et de facilité d'utilisation.

2. les tests à réaliser conformément aux plans et aux procédures ;
3. les moyens et organisations intervenant dans la réalisation de la fonction d'assurance lors des tests et de la validation.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR, CDR].

6.3.5.2

- a. En fonction de la criticité des logiciels, les objectifs de couverture de test pour chaque niveau de test doivent être convenus entre le client et le fournisseur et leur réalisation doit être contrôlée par des métriques pour :

1. les tests unitaires ;
2. les tests d'intégration ;
3. les essais de validation par rapport à la spécification technique et par rapport aux exigences de besoin.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR, CDR].

6.3.5.3

- a. Le fournisseur doit garantir, via une revue interne, que les procédures et les données de test sont correctes, réalisables et traçables, et qu'elles satisfont aux exigences.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.3.5.4

- a. Des revues d'aptitude aux tests doivent avoir été tenues avant le début des activités de test, comme défini dans le plan de développement logiciel.

RESULTAT ATTENDU : rapports de revue d'aptitude au test [DJF, -; ARE].

6.3.5.5

- a. La couverture des tests doit être vérifiée par rapport aux objectifs définis.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

- b. Des retours sur les résultats de l'évaluation de la couverture des tests doivent être faits en continu aux développeurs.

6.3.5.6

- a. Le fournisseur doit s'assurer que les non conformités et les problèmes détectés durant les tests sont correctement documentés et transmis aux personnes concernées.

RESULTAT ATTENDU : fiches d'anomalies et problème logiciel [DJF, -; CDR, QR, AR, ORR].

6.3.5.7

- a. La couverture des tests pour le code configurable doit être contrôlée pour s'assurer que les exigences définies sont respectées dans chaque configuration testée.

RESULTAT ATTENDU : déclaration de conformité aux plans et aux procédures de test [PAF, -; CDR, QR, AR, ORR].

6.3.5.8

- a. La réalisation des actions associées aux problèmes logiciel générés au cours des tests et de la validation doit être vérifiée et enregistrée.

RESULTAT ATTENDU : problèmes logiciel [DJF, -; SRR, PDR, CDR, QR, AR, ORR].

6.3.5.9

- a. Des dispositions doivent être prises pour permettre au client d'assister aux tests.

6.3.5.10

- a. Des dispositions doivent être prises pour permettre à du personnel fournisseur indépendant de l'équipe de développement d'assister aux tests.

NOTE Par exemple : personnel spécialiste de l'assurance produit logiciel.

6.3.5.11

- a. Le fournisseur doit s'assurer que :
1. les tests sont menés à l'aide de procédures et de données de test approuvées ;
 2. la configuration soumise aux tests est correcte ;
 3. les tests sont correctement documentés ;
 4. les rapports de test sont à jour et valides.

RESULTAT ATTENDU : déclaration de conformité aux plans et aux procédures de test [PAF, -; CDR, QR, AR, ORR].

6.3.5.12

- a. Le fournisseur doit s'assurer que les tests sont rejouables en vérifiant l'archivage du logiciel testé, des logiciels support, de l'environnement d'essai, des documents support et des problèmes rencontrés.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

6.3.5.13

- a. Le fournisseur doit confirmer par écrit que les tests ont été réalisés avec succès.

RESULTAT ATTENDU : rapports de test et de validation [DJF, -; CDR, QR, AR, ORR].

6.3.5.14

- a. Des commissions de revue chargées d'examiner les aspects ingénierie et assurance produit doivent se réunir après la fin des phases de test, conformément au plan de développement logiciel.

6.3.5.15

- a. Les zones concernées par une modification doivent être identifiées et soumises à de nouveaux tests (tests de non régression).

6.3.5.16

- a. En cas de rejeu des tests, toute la documentation relative à ces tests (procédures, données et rapports d'essai) doit être mise à jour.

RESULTAT ATTENDU : documentation de test mise à jour [DJF, -; CDR, QR, AR, ORR].

6.3.5.17

- a. La nécessité de soumettre le logiciel à des tests de non régression et à des vérifications complémentaires doit être analysée après toute évolution matérielle de la plate-forme.

RESULTAT ATTENDU : documentation de test mise à jour [DJF, -; CDR, QR, AR, ORR].

6.3.5.18

- a. La nécessité de soumettre le logiciel à des tests de non régression et à des vérifications complémentaires doit être analysée après toute évolution ou mise à jour de tout outil utilisé pour sa génération.

NOTE Par exemple : code source ou code objet.

RESULTAT ATTENDU : documentation de test mise à jour [DJF, -; CDR, QR, AR, ORR].

6.3.5.19

- a. La validation doit être effectuée par une équipe n'ayant pas pris part à la conception ou au codage du logiciel à valider.

NOTE Ceci peut être réalisé au niveau de l'ensemble du produit logiciel ou bien composant par composant.

6.3.5.20

- a. La validation du logiciel de vol par rapport aux exigences de besoin sur le modèle de vol doit être réalisée sur une version du logiciel exempte de patch.

6.3.5.21

- a. Le fournisseur doit passer en revue la documentation de test afin de s'assurer qu'elle est à jour et qu'elle est organisée de façon à faciliter sa réutilisation en phase de maintenance.

6.3.5.22

- a. Les tests doivent être organisés comme des activités à part entière, tant en termes de planification, que de ressources et de composition des équipes.

RESULTAT ATTENDU : documentation de test et de validation [DJF, SValP; PDR], [DJF, SUITP; PDR, CDR].

6.3.5.23

- a. Les ressources nécessaires aux tests doivent être identifiées au début du cycle de vie en tenant compte des exigences d'exploitation et de maintenance.

RESULTAT ATTENDU : documentation de test et de validation [DJF, SValP; PDR], [DJF, SUITP; PDR, CDR].

6.3.5.24

- a. Le développement ou l'acquisition d'outils de test (matériel et logiciel) doivent être prévus dans le cadre de la planification globale du projet.

RESULTAT ATTENDU : documentation de test et de validation [DJF, SValP; PDR], [DJF, SUITP; PDR, CDR].

6.3.5.25

- a. Le fournisseur doit établir et passer en revue les procédures et données de test avant le début des activités d'essai et doit également documenter les contraintes des essais en ce qui concerne les limitations physiques, de performances, fonctionnelles, de contrôle et d'observabilité.

RESULTAT ATTENDU : documentation de test et de validation [DJF, SValP; PDR], [DJF, SVS; CDR, QR, AR], [DJF, SUITP; PDR, CDR].

6.3.5.26

- a. Avant de présenter le produit pour livraison et recette par le client, le fournisseur doit valider son fonctionnement en tant que produit complet, dans des conditions similaires à celles de son environnement d'exploitation, conformément aux exigences de besoin.

6.3.5.27

- a. Quand des tests sont réalisés dans l'environnement d'exploitation, les points suivants doivent être abordés :
1. les fonctionnalités à tester dans l'environnement d'exploitation ;
 2. les responsabilités particulières du fournisseur et du client dans l'exécution et l'évaluation des résultats de test ;
 3. restauration de l'environnement d'exploitation initial (après les tests).

RESULTAT ATTENDU : documentation de test et de validation [DJF, - ; AR].

6.3.5.28

- a. Une validation indépendante du logiciel doit être réalisée par un tiers.

NOTE Cette exigence n'est applicable que lorsque les risques associés au projet justifient les coûts induits. Le client peut envisager un niveau d'indépendance moindre, par exemple une équipe indépendante au sein de la même entité.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. plan ISVV [DJF, -; SRR, PDR] ;
- b. rapport ISVV [DJF, -; PDR, CDR, QR, AR].

6.3.5.29

- a. La validation doit comprendre des essais dans les différentes configurations possibles ou dans un ensemble représentatif de ces configurations lorsque le nombre de configurations possibles est manifestement trop élevé pour permettre de toutes les valider.

RESULTAT ATTENDU : documentation de test et de validation [DJF, SValP; PDR], [DJF, SVS; CDR, QR, AR].

6.3.5.30

- a. Les logiciels contenant du code désactivé doivent être validés spécifiquement pour garantir que le code désactivé ne peut pas être activé ou que son activation accidentelle ne peut pas nuire au fonctionnement du système.

RESULTAT ATTENDU : rapports de test et de validation [DJF, -; CDR, QR, AR].

6.3.5.31

- a. Les logiciels contenant du code configurable doivent être validés spécifiquement pour garantir qu'une configuration non désirée ne peut pas être activée lors de l'exécution du logiciel ni être prise en compte lors de la génération du code.

RESULTAT ATTENDU : rapports de test et de validation [DJF, -; CDR, QR, AR].

6.3.5.32

- a. Les activités de validation des exigences qualité doivent être spécifiées dans la définition de la spécification de validation.

RESULTAT ATTENDU : spécification de validation du logiciel [DJF, SVS; CDR, QR, AR].

6.3.6 Livraison et recette des logiciels

6.3.6.1

- a. Les rôles, responsabilités et obligations du fournisseur et du client lors de l'installation doivent être définis.

RESULTAT ATTENDU : procédure d'installation [DDF, SCF; AR].

6.3.6.2

- a. L'installation doit être réalisée conformément à la procédure d'installation.

6.3.6.3

- a. Le client doit établir un plan de test de recette définissant les tests de recette prévus, y compris les tests spécifiques à l'environnement cible (voir le paragraphe 5.7.3.1 de l'ECSS-E-ST-40).

NOTE 1 Les tests de recette peuvent être en partie constitués de tests utilisés lors d'activités de test précédentes.

NOTE 2 Le plan de test de recette tient compte de l'exigence de démonstration opérationnelle soit pendant la recette, soit après celle-ci.

RESULTAT ATTENDU : plan de test de recette [DJF, -; QR, AR].

6.3.6.4

- a. Le client doit s'assurer que les tests de recette son menés conformément au plan de test de recette approuvé (voir le paragraphe 5.7.3.2 de l'ECSS-E-ST-40).

6.3.6.5

- a. Avant de présenter le logiciel au client pour recette, le fournisseur doit s'assurer que :
 1. le logiciel livré est conforme aux exigences contractuelles (y compris tout contenu spécifié par la documentation de recette) ;
 2. le code objet fourni correspond au code source fourni ;
 3. toutes les modifications acceptées ont été mises en œuvre ;
 4. toutes les non-conformités sont soit résolues, soit déclarées.

6.3.6.6

- a. Le client doit vérifier que le code exécutable a été régénéré à partir de composants de code source gérés en configuration et installé conformément à des procédures prédéfinies dans l'environnement cible.

6.3.6.7

- a. Tous les problèmes rencontrés doivent être documentés dans des fiches d'anomalies.

RESULTAT ATTENDU : fiches d'anomalies [DJF, -; AR].

6.3.6.8

- a. A la fin des tests de recette, un compte rendu doit être rédigé et signé par les représentants du fournisseur et du client, les ingénieurs qualité logiciel des deux parties et le représentant de l'entité en charge de la maintenance du produit logiciel.

RESULTAT ATTENDU : rapport de test de recette [DJF, -; AR].

6.3.6.9

- a. Le client doit certifier la conformité aux procédures et établir la conclusion relative aux résultats de test du produit logiciel testé (accepté, accepté avec réserve, refusé).

RESULTAT ATTENDU : rapport de test de recette [DJF, -; AR].

6.3.7 Opérations

6.3.7.1

- a. Pendant les opérations, la qualité des produits de mission liés au logiciel doit être convenue entre le client et les utilisateurs.

NOTE : La qualité des produits de mission peut comprendre des paramètres tels que : données exemptes d'erreurs, disponibilité des données et écarts acceptables ; dégradation acceptable des informations.

RESULTAT ATTENDU : plan support d'exploitation du logiciel [OP, -; ORR].

6.3.7.2

- a. Au cours de la démonstration de la conformité du logiciel aux exigences opérationnelles, au minimum, les points suivants doivent être abordés :
1. disponibilité et maintenabilité du système hôte (notamment en ce qui concerne le redémarrage après des interventions de maintenance) ;
 2. fonctionnalités de sécurité ;
 3. interface homme-machine ;
 4. procédures d'exploitation ;
 5. conformité aux exigences qualité des produits de mission.

RESULTAT ATTENDU : validation des exigences opérationnelles [PAF, -; ORR].

6.3.7.3

- a. Le plan assurance produit relatif à l'exploitation du système doit prendre en considération le logiciel.

RESULTAT ATTENDU : entrée pour le plan assurance produit relatif à l'exploitation des systèmes [PAF, -; ORR].

6.3.8 Maintenance

6.3.8.1

- a. L'entité responsable de la maintenance doit être identifiée pour faciliter la transition vers les opérations et la maintenance.

Note Une équipe comprenant des représentants du fournisseur et du client peut être constituée pour assurer les activités de maintenance. Une attention particulière est accordée à l'importance de la souplesse de cette équipe pour faire face au caractère aléatoire des problèmes et à l'identification des moyens et ressources à utiliser pour les activités de maintenance.

RESULTAT ATTENDU : plan de maintenance [MF, -; QR, AR, ORR].

6.3.8.2

- a. L'entité chargée de la maintenance doit spécifier les activités d'assurance, de vérification et de validation applicables aux interventions de maintenance.

RESULTAT ATTENDU : plan de maintenance [MF, -; QR, AR, ORR].

6.3.8.3

- a. Les plans de maintenance doivent être vérifiés par rapport aux exigences spécifiées pour la maintenance du produit logiciel.

NOTE Les plans et procédures de maintenance peuvent évoquer la maintenance corrective, évolutive, adaptative et préventive, en différenciant les activités de maintenance « de routine » et les activités de maintenance « d'urgence ».

6.3.8.4

- a. Les plans et procédures de maintenance doivent comprendre au minimum les éléments suivants :

1. objectif de la maintenance ;
2. identification de la première version du produit logiciel pour laquelle la maintenance doit être effectuée ;
3. organisation support ;
4. cycle de vie de la maintenance ;
5. activités de maintenance ;
6. dispositions qualité applicables à la maintenance ;
7. enregistrements et rapports de maintenance.

RESULTAT ATTENDU : plan de maintenance [MF, -; QR, AR, ORR].

6.3.8.5

- a. Pour la soumission des rapports de maintenance, des règles doivent être établies et intégrées au plan de maintenance.

RESULTAT ATTENDU : plan de maintenance [MF, -; QR, AR, ORR].

6.3.8.6

- a. Toutes les activités de maintenance doivent être enregistrées dans des formats prédéfinis et conservées.

RESULTAT ATTENDU : enregistrements de maintenance [MF, -; -].

6.3.8.7

- a. Des enregistrements de maintenance doivent être établis pour chaque produit logiciel en incluant au minimum les informations suivantes :

1. liste des demandes d'assistance ou des problèmes reçus et état actuel de chacun d'eux ;
2. entité chargée de répondre aux demandes d'assistance ou de mettre en œuvre les actions correctives appropriées ;
3. priorités assignées aux actions correctives ;
4. résultats des actions correctives ;
5. données statistiques sur les défaillances survenues et les activités de maintenance.

NOTE L'enregistrement des activités de maintenance peut être utilisé pour l'évaluation et l'amélioration du produit logiciel et pour l'amélioration du système qualité lui-même.

RESULTAT ATTENDU : enregistrements de maintenance [MF, -; -].

Assurance qualité du produit logiciel

7.1 Objectifs qualité produit et métriques

7.1.1 Déclinaison des exigences

- a. Les exigences qualité logiciel (notamment les exigences de sécurité et de sûreté de fonctionnement) doivent être dérivées des exigences définies au niveau système.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. exigences de besoin [RB, SSS; SRR] ;
- b. spécification technique [TS, SRS; PDR].

7.1.2 Définition quantitative des exigences qualité

- a. Les exigences qualité doivent être exprimées en termes de quantités ou de contraintes.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. exigences de besoin [RB, SSS; SRR] ;
- b. spécification technique [TS, SRS; PDR].

7.1.3 Activités d'assurance relatives aux exigences qualité produit

- a. Le fournisseur doit définir des activités d'assurance permettant de garantir que le produit répond aux exigences qualité définies dans la spécification technique.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

7.1.4 Métriques sur les produits

- a. Pour vérifier la mise en œuvre des exigences qualité produit, le fournisseur doit définir un programme de métriques s'appuyant sur le modèle qualité identifié (voir le paragraphe 5.2.7) et précisant :
1. les métriques à collecter et à conserver ;

2. les moyens utilisés pour collecter les métriques (mesures) ;
3. les valeurs cibles en relation avec les exigences qualité produit ;
4. les analyses à réaliser sur les métriques collectées, notamment celles destinées à permettre de déduire :

(a) des statistiques descriptives ;

NOTE Par exemple : le nombre d'unités appartenant à chaque niveau de complexité.

(b) des analyses de tendances (par exemple, tendances relatives aux problèmes logiciel).

5. la façon dont les résultats des analyses réalisées sur les métriques collectées sont communiqués à l'équipe de développement et utilisés pour identifier des actions correctives ;
6. le calendrier de collecte, de stockage, d'analyse et de diffusion des métriques, se référant au cycle de vie complet du logiciel.

NOTE L'ECSS-Q-HB-80-04 contient des conseils concernant la mise en œuvre du programme de métriques logiciel.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

7.1.5 Métriques de base

- a. Les métriques produit de base suivantes doivent être utilisées :
 1. taille (code) ;
 2. complexité (conception, code) ;
 3. taux de fautes et de défaillances ;
 4. couverture des tests ;
 5. nombre de défaillances.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; SRR, PDR].

7.1.6 Rapports sur les métriques

- a. Les résultats de la collecte et de l'analyse des métriques doivent être inclus dans les bilans qualité logiciel, le but étant d'offrir au client un aperçu du niveau de qualité obtenu.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

7.1.7 Précision numérique

- a. La précision numérique doit être estimée et vérifiée.

RESULTAT ATTENDU : analyse de la précision numérique [DJF, SVR; PDR, CDR, QR].

7.1.8 Analyse de la maturité du logiciel

- a. Le fournisseur doit définir l'organisation et les moyens mis en œuvre pour collecter et analyser les données nécessaires à l'étude de la maturité du logiciel.

NOTE Par exemple : défaillances, corrections, durée des exécutions.

RESULTAT ATTENDU : bilans qualité logiciel [PAF, -; -].

7.2 Exigences qualité produit

7.2.1 Exigences de besoin et spécification technique

7.2.1.1

- a. Les exigences qualité logiciel doivent être documentées dans les exigences de besoin et dans la spécification technique.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *exigences de besoin [RB, SSS; SRR] ;*
- b. *spécification technique [TS, SRS; PDR].*

7.2.1.2

- a. Les exigences logiciel doivent être :

1. correctes ;
2. univoques ;
3. complètes ;
4. cohérentes ;
5. vérifiables ;
6. traçables.

7.2.1.3

- a. Pour chaque exigence, la méthode de vérification et de validation doit être spécifiée.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *exigences de besoin [RB, SSS; SRR] ;*
- b. *spécification technique [TS, SRS; PDR].*

7.2.2 Conception et documentation associée

7.2.2.1

- a. La conception logiciel doit satisfaire aux exigences non fonctionnelles telles que documentées dans la spécification technique.

7.2.2.2

- a. Les logiciels doivent être conçus pour faciliter les tests.

7.2.2.3

- a. Les logiciels prévus pour avoir une longue durée de vie doivent être conçus de sorte à être le moins dépendants possible vis-à-vis du système d'exploitation et du matériel, l'objectif étant de favoriser leur portabilité.

NOTE Cette exigence s'applique aux situations pour lesquelles la durée de vie du logiciel peut mener à l'obsolescence et à l'indisponibilité du système d'exploitation et/ou du matériel d'origine et menacer sa maintenabilité.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *plan assurance produit logiciel [PAF, SPAP; SRR, PDR] ;*
- b. *justification des choix de conception [DDF, SDD; PDR, CDR].*

7.2.3 Documentation de test et de validation

7.2.3.1

- a. La documentation de test et de validation détaillée (données, procédures et résultats attendus) définie dans le DJF de l'ECSS-E-ST-40 doit être cohérente par rapport à la stratégie de test et de validation définie (voir le paragraphe 6.3.5 et les paragraphes 5.5.3, 5.5.4, 5.6 et 5.8 de l'ECSS-E-ST-40).

7.2.3.2

- a. La documentation de test doit couvrir l'environnement de test, les outils et les logiciels de test, le personnel nécessaire et les exigences de formation associées.

7.2.3.3

- a. Les critères d'achèvement de chaque test et de toute éventuelle étape doivent être spécifiés.

7.2.3.4

- a. Les procédures de test, les données et les résultats attendus doivent être spécifiés.

7.2.3.5

- a. La configuration matérielle et logicielle doit être identifiée et documentée dans la documentation de test.

7.2.3.6

- a. Pour toutes les exigences non couvertes par les essais, un rapport de vérification documentant ou référant les activités de vérification effectuées doit être rédigé.

RESULTAT ATTENDU : rapport de vérification logiciel [DJF, SVR; CDR, QR, AR].

7.3 Logiciels destinés à être réutilisés

7.3.1 Exigences du client

- a. Pour le développement de logiciels destinés à être réutilisés, les paragraphes 5.2.4.7 et 5.4.3.6 de l'ECSS-E-ST-40 doivent être appliqués.

7.3.2 Documentation séparée

- a. Les informations relatives aux composants développés en vue d'être réutilisés doivent être séparées des autres dans la spécification technique, le dossier justificatif de définition, le dossier de conception et le dossier d'assurance produit.

7.3.3 Informations autosuffisantes

- a. Les informations relatives aux composants développés en vue d'être réutilisés se trouvant dans la spécification technique, le dossier justificatif de définition, le dossier de conception et le dossier d'assurance produit doivent être autosuffisantes.

7.3.4 Exigences relatives aux réutilisations prévues

- a. La spécification technique des composants développés en vue d'être réutilisés doit comporter les exigences de maintenabilité, de portabilité et de vérification relatives auxdits composants.

RESULTAT ATTENDU : spécification technique des composants réutilisables [TS, -; PDR].

7.3.5 Gestion de configuration en vue d'une réutilisation

- a. Le système de gestion de configuration doit inclure des dispositions relatives aux aspects spécifiques des logiciels développés en vue d'être réutilisés, tels que :
 - la durée de vie supérieure des composants développés en vue d'être réutilisés par rapport aux autres composants du projet ;
 - l'évolution ou la modification de l'environnement de développement pour le futur projet dans lequel l'utilisation de ces composants est envisagée ;
 - le transfert des informations relatives à la gestion de configuration et de documentation vers le futur projet pour lequel une réutilisation du logiciel est prévue.

RESULTAT ATTENDU : plan de gestion de configuration des logiciels [MGT, SCMP; SRR, PDR].

7.3.6 Tests sur plusieurs plates-formes

- a. Lorsque des composants sont développés de manière à être réutilisés sur différentes plates-formes, les tests de ces composants doivent être réalisés sur toutes ces plates-formes.

RESULTAT ATTENDU : documentation de vérification et de validation des composants réutilisables [DJF, -; CDR].

7.3.7 Certificat de conformité

- a. Le fournisseur doit présenter un certificat de conformité attestant que les tests ont été réalisés avec succès sur toutes les plates-formes correspondantes.

NOTE En cas d'indisponibilité de certaines plates-formes, le certificat de conformité indique les limites des tests réalisés.

RESULTAT ATTENDU : documentation de vérification et de validation des composants réutilisables [DJF, -; CDR].

7.4 Matériel et services au sol standard pour un système opérationnel

7.4.1 Acquisition de matériel

- a. La sous-traitance et l'acquisition de matériel doivent être menées selon les exigences du paragraphe 5.4 de l'ECSS-Q-ST-20.

RESULTAT ATTENDU : les résultats suivants sont attendus :

- a. *justification de la sélection des équipements au sol opérationnels [DJF, -; SRR, PDR] ;*
b. *comptes rendus des contrôles à la réception [PAF, -; SRR, PDR].*

7.4.2 Acquisition de services

- a. L'acquisition de services de support à utiliser lors des phases d'exploitation doit être justifiée comme couvrant les contrats de niveau de service, la qualité de service et les procédures d'escalade requis pour l'exploitation et la maintenance du système.

RESULTAT ATTENDU : justification de la sélection des services de support opérationnels [DJF, -; SRR, PDR].

7.4.3 Contraintes

- a. Le choix du matériel et des services acquis doit tenir compte des contraintes associées à la fois au développement et à l'utilisation réelle du logiciel.

RESULTAT ATTENDU : justification de la sélection des équipements au sol opérationnels [DJF, -; SRR, PDR].

7.4.4 Sélection

- a. Le matériel informatique et les services de support au sol prévus pour la mise en œuvre du système final doivent être sélectionnés en fonction des exigences du projet concernant :

1. les performances ;
2. la maintenance ;
3. la pérennité et la cohérence technique vis-à-vis des équipements opérationnels ;
4. l'évaluation du produit par rapport aux exigences, y compris la catégorie de criticité ;
5. la disponibilité de la documentation de soutien ;
6. les conditions de recette et de garantie ;
7. les conditions d'installation, de préparation, de formation et d'utilisation ;
8. les conditions de maintenance, y compris les possibilités d'évolutions ;
9. les contraintes de copyright ;
10. la disponibilité ;
11. la compatibilité ;
12. les contraintes d'exploitation du site.

RESULTAT ATTENDU : justification de la sélection des équipements au sol opérationnels [DJF, -; SRR, PDR].

7.4.5 Maintenance

- a. En tenant compte de la politique du fabricant pour la maintenance et le produit, il faut garantir que le matériel et les services de support peuvent être maintenus tout au long de la durée de vie spécifiée pour le produit logiciel dans le cadre des contraintes d'exploitation.

7.5 Microprogrammes

7.5.1 Programmation des dispositifs

- a. Le fournisseur doit établir des procédures pour la programmation et la duplication des dispositifs contenant des microprogrammes.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR].

7.5.2 Marquage

- a. Les dispositifs contenant des microprogrammes doivent être marqués de manière indélébile pour permettre l'identification (par référence) du composant matériel et du composant logiciel.

RESULTAT ATTENDU : plan assurance produit logiciel [PAF, SPAP; PDR].

7.5.3 Etalonnage

- a. Le fournisseur doit garantir que l'équipement de programmation du microprogramme est étalonné.

Annexe A (informative) Documentation logiciel

La présente annexe définit la structure des documents logiciel à produire, illustrée par la Figure A-1.

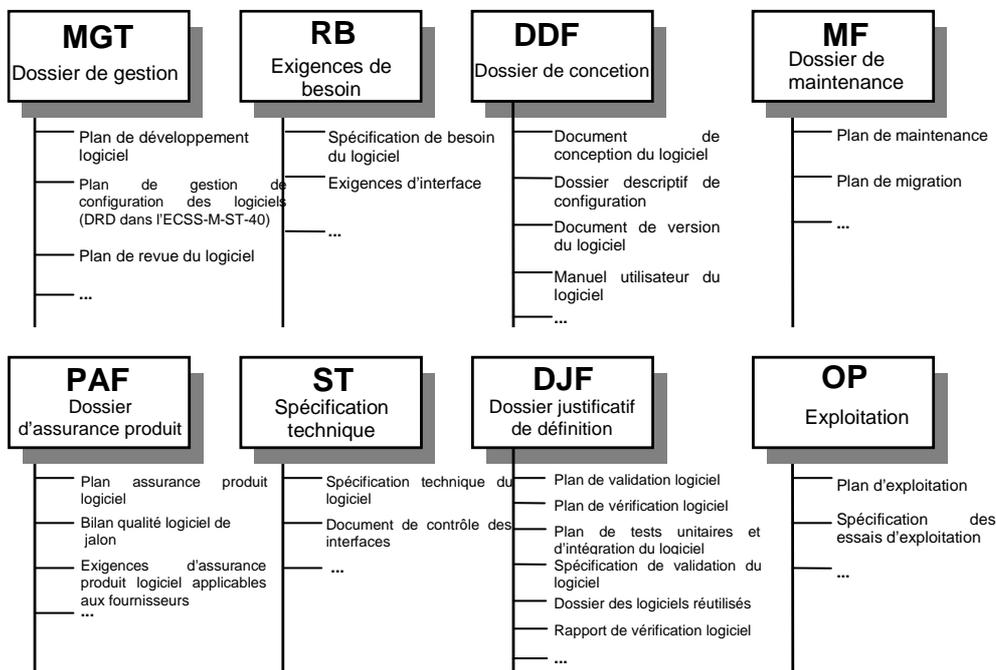


Figure A-1: Présentation générale des documents logiciel

Le Tableau A-1 présente la liste des exigences documentaires et identifie la documentation logiciel à produire en fonction des exigences définies dans ce standard et dans l'ECSS-E-ST-40.

Tableau A-1: Liste des exigences documentaires (SRF) des standards ECSS-E-ST-40 et ECSS-Q-ST-80

Docu-ment associé	Documentation (ex. : plan, document, dossier, rapport, formulaire, matrice)	DRD	SRR	PDR	CDR	QR	AR	ORR
RB	Spécification de besoin du logiciel (SSS)	ECSS-E-ST-40 Annexe B	✓					
	Exigences d'interfaces (IRD)	ECSS-E-ST-40 Annexe C	✓					
	Résultats de l'analyse de sécurité et de sûreté de fonctionnement des fournisseurs de niveau inférieur	-	✓					
ST	Spécification technique du logiciel (SRS)	ECSS-E-ST-40 Annexe D		✓				
	Document de contrôle des interfaces logiciel (ICD)	ECSS-E-ST-40 Annexe E		✓	✓			
DD	Document de conception du logiciel (SDD)	ECSS-E-ST-40 Annexe F		✓	✓			
	Dossier de configuration du logiciel (SCF)	ECSS-M-ST-40 Annexe E		✓	✓	✓	✓	✓
	Document de version du logiciel (SRelD)	ECSS-E-ST-40 Annexe G				✓	✓	
	Manuel utilisateur du logiciel (SUM)	ECSS-E-ST-40 Annexe H			✓	✓	✓	
	Code source du logiciel et marquage des supports	-			✓			
	Produit logiciel et marquage des supports	-				✓	✓	✓
	Matériel de formation	-				✓		
DJF	Plan de vérification logiciel (SVerP)	ECSS-E-ST-40 Annexe I		✓				
	Plan de validation logiciel (SValP)	ECSS-E-ST-40 Annexe J		✓				

Docu- ment associé	Documentation (ex. : plan, document, dossier, rapport, formulaire, matrice)	DRD	SRR	PDR	CDR	QR	AR	ORR
	Plan de vérification et de validation indépendante des logiciels	-	✓	✓				
	Plan de tests d'intégration du logiciel (SUITP)	ECSS-E-ST-40C Annexe K		✓	✓			
	Plan de test unitaires du logiciel (SUITP)	ECSS-E-ST-40 Annexe K			✓			
	Spécification de validation du logiciel (SVS) par rapport à la TS	ECSS-E-ST-40 Annexe L			✓			
	Spécification de validation du logiciel (SVS) par rapport aux RB	ECSS-E-ST-40 Annexe L				✓	✓	
	Plan de test de recette	-				✓	✓	
	Rapport de tests unitaires du logiciel	-			✓			
	Rapport de test d'intégration du logiciel	-			✓			
	Rapport de validation du logiciel par rapport à la TS	-			✓			
	Rapport de validation du logiciel par rapport aux RB	-				✓	✓	
	Rapport de test de recette	-					✓	
	Rapport d'installation	-					✓	
	Rapport de vérification logiciel (SVR)	ECSS-E-ST-40 Annexe M	✓	✓	✓	✓	✓	✓
	Rapport de vérification et de validation indépendantes des logiciels	-		✓	✓	✓	✓	✓
	Dossier des logiciels réutilisés (SRF)	ECSS-E-ST-40 Annexe N	✓	✓	✓			

Docu- ment associé	Documentation (ex. : plan, document, dossier, rapport, formulaire, matrice)	DRD	SRR	PDR	CDR	QR	AR	ORR
	Fiches d'anomalies et problème	-	✓	✓	✓	✓	✓	✓
	Rapport de revue commune	-	✓	✓	✓	✓	✓	
	Justification de la sélection des équipements et services au sol opérationnels	-	✓	✓				
MGT	Plan de développement logiciel (SDP)	ECSS-E-ST-40 Annexe O	✓	✓				
	Plan de revue du logiciel (SRevP)	ECSS-E-ST-40 Annexe P	✓	✓				
	Plan de gestion de configuration des logiciels	ECSS-M-ST-40 Annexe A	✓	✓				
	Plan de formation	-	✓					
	Procédures de gestion des interfaces	-	✓					
	Identification des membres de la commission de traitement des anomalies logiciel	-	✓					
	Données d'approvisionnement	-	✓	✓				
MF	Plan de maintenance	-				✓	✓	✓
	Enregistrements de maintenance	-				✓	✓	✓
	Problèmes et fiches d'anomalie - Rapport d'analyse des modifications - Rapport d'analyse des anomalies - Identification des modifications	-						
	Plan de migration et notification	-						
	Plan de retrait de service et notification	-						

Docu- ment associé	Documentation (ex. : plan, document, dossier, rapport, formulaire, matrice)	DRD	SRR	PDR	CDR	QR	AR	ORR
OP	Plan support d'exploitation du logiciel	-						✓
	Résultats des essais d'exploitation	-						✓
	Problèmes et fiches d'anomalie - Produit logiciel d'enregistrement des demandes de l'utilisateur - Rapport de revue après exploitation	-						✓
PAF	Plan d'assurance produit logiciel (SPAP)	ECSS-Q-ST-80 Annexe B	✓	✓	✓	✓	✓	✓
	Exigences d'assurance produit logiciel applicables aux fournisseurs	-	✓					
	Plan et calendrier des audits	-	✓					
	Plans ou procédures de revue et de contrôle	-						
	Procédures et standards	-		✓				
	Standards de modélisation et de conception	-	✓	✓				
	Standards de codage et description des outils	-		✓				
	Procédure de gestion des anomalies logiciel	-		✓				
	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels - Classification de la criticité des composants logiciel	-		✓	✓	✓	✓	
	Bilans qualité logiciel	-						
Bilans qualité logiciel de jalon (SPAMR)	ECSS-Q-ST-80 Annexe C	✓	✓	✓	✓	✓	✓	

Docu- ment associé	Documentation (ex. : plan, document, dossier, rapport, formulaire, matrice)	DRD	SRR	PDR	CDR	QR	AR	ORR
	Déclaration de conformité aux plans et aux procédures de test	-			✓	✓	✓	✓
	Justificatifs relatifs à la formation et à l'expérience	-						
	Avis d'alerte (préliminaire)	-						
	Résultats des audits et des évaluations avant contrat et des sources d'approvisionnement	-						
	Plan d'évaluation des processus logiciel	-						
	Enregistrements des évaluations des processus logiciel	-						
	Rapports de revue et de contrôle	-						
	Comptes rendus des contrôles à la réception	-	✓	✓	✓	✓		
	Entrée pour le plan assurance produit relatif à l'exploitation des systèmes	-						✓

Annexe B (normative)

Plan assurance produit logiciel (SPAP) - DRD

B.1 Identification de la DRD

B.1.1 Identification des exigences et document source

Le plan assurance produit logiciel (SPAP) est demandé par les exigences normatives résumées dans le Tableau B-1.

Tableau B-1 : Traçabilité du SPAP par rapport aux paragraphes des standards ECSS-E-ST-40 et ECSS-Q-ST-80

Standard ECSS	Paragraphe	Section de la DRD
ECSS-Q-ST-80	5.1.2.1	<5.1>, <5.2>
	5.1.2.2	<5.1>, <5.2>
	5.1.2.3	<5.1>
	5.1.3.1	<5.3>
	5.1.4.1	<5.2>
	5.2.1.1	Toutes
	5.2.1.3	Toutes
	5.2.1.4	<6.7>g
	5.2.1.5	<8>
	5.2.6.2	<6.4>
	5.2.7.2	<5.5>
	5.4.3.3	Toutes
	5.4.3.4	Toutes
	5.6.1.1	<5.8>
	6.1.1	<6.1>
	6.1.5	<6.1>c
	6.2.1.4	<6.2>
	6.2.3.1	<6.3>
	6.2.3.2	<6.3>

Standard ECSS	Paragraphe	Section de la DRD
	6.2.3.4	<6.3>
	6.2.3.5	<6.3>
	6.2.4.8	<6.4>
	6.2.4.9	<6.4>
	6.2.4.11	<6.4>
	6.2.5.1	<6.5>
	6.2.5.2	<6.5>
	6.2.7.2	<6.6>
	6.2.7.3	<6.6>
	6.2.7.4	<6.6>
	6.2.7.5	<6.6>
	6.3.3.3	<6.8>.c
	6.3.3.5	<6.7>.b
	6.3.3.7	<6.7>.b
	6.3.4.3	<6.8>.b
	6.3.4.6	<6.7>.c
	6.3.5.1	<6.7>.d
	6.3.5.2	<6.7>.d
	7.1.3	<7.2>.d
	7.1.5	<7.2>
	7.1.6	<7.2>
	7.2.2.3	<7.1>
	7.5.1	<6.8>.c
	7.5.2	<6.8>.c

B.1.2 Objet

Le plan d'assurance produit logiciel fait partie du dossier d'assurance produit (PAF).

Le plan d'assurance produit logiciel vise à décrire les aspects organisationnels et à présenter l'approche technique utilisée pour l'exécution du programme d'assurance produit logiciel.

B.2 Réponse attendue

B.2.1 Domaine d'application et contenu

<1> Introduction

- a. Le SPAP doit contenir une description du but, de l'objectif, du contenu et de la raison de son élaboration.

<2> Documents applicables et de référence

- a. Le SPAP doit répertorier les documents applicables et de référence utilisés en support de sa rédaction.

<3> Termes, définitions et abréviations

- a. Le SPAP doit inclure tous les autres termes, définitions et abréviations utilisés.

<4> Présentation du système

- a. Le SPAP doit comporter une description du système et des produits logiciel à développer, ou y faire référence.

<5> Mise en œuvre du programme d'assurance produit logiciel

<5.1> Organisation

- a. Le SPAP doit décrire l'organisation des activités d'assurance produit logiciel, et notamment les responsabilités, l'autorité et les relations mutuelles du personnel chargé de la gestion, de l'exécution et de la vérification des travaux ayant une incidence sur la qualité du logiciel.
- b. Les points suivants doivent être abordés :
1. structure organisationnelle ;
 2. interfaces de chaque organisation, externe ou interne, impliquée dans le projet ;
 3. relations avec l'assurance produit et la sécurité de niveau système ;
 4. indépendance de la fonction d'assurance produit logiciel ;
 5. délégation des tâches d'assurance produit logiciel à un fournisseur de niveau inférieur, le cas échéant.

<5.2> Responsabilités

- a. Le SPAP doit décrire les responsabilités de la fonction d'assurance produit logiciel.

<5.3> Ressources

- a. Le SPAP doit décrire les ressources à utiliser pour réaliser la fonction d'assurance produit logiciel.
- b. La description du paragraphe B.2.1<5.3>a. doit inclure les ressources humaines, les compétences, le matériel et les outils logiciel.

<5.4> Rapports

- a. Le SPAP doit décrire les rapports à réaliser par l'assurance produit logiciel.

<5.5> Modèles qualité

- a. Le SPAP doit décrire les modèles qualité applicables au projet et la façon dont ceux-ci sont utilisés pour définir les exigences qualité.

<5.6> Management des risques

- a. Le SPAP doit décrire la contribution de la fonction d'assurance produit logiciel au management des risques du projet.

<5.7> Sélection et suivi des fournisseurs

- a. Le SPAP doit décrire la contribution de la fonction d'assurance produit logiciel à la sélection et au suivi des fournisseurs de niveau immédiatement inférieur.

<5.8> Méthodes et outils

- a. Le SPAP doit décrire les méthodes et les outils utilisés pour toutes les activités du cycle de développement, ainsi que leur niveau de maturité.

<5.9> Evaluation et amélioration des processus

- a. Le SPAP doit définir le domaine d'application et les objectifs de l'évaluation des processus.
- b. Le SPAP doit décrire les méthodes et outils à utiliser lors de l'évaluation et de l'amélioration des processus.

<5.10> Exploitation et maintenance (facultatif)

- a. Le SPAP doit définir les dispositions qualité relatives aux processus d'exploitation et de maintenance (il est également possible de créer un SPAP indépendant).

<6> Assurance des processus logiciel

<6.1> Cycle de développement logiciel

- a. Le SPAP doit faire référence à la description du cycle de développement logiciel présenté dans le plan de développement logiciel.
- b. Si le cycle de vie n'est pas présenté dans le plan de développement logiciel, il doit être décrit dans le SPAP.
- c. Le cycle de vie doit comporter un jalon situé juste avant le début de la validation du logiciel.

<6.2> Plans de projet

- a. Le SPAP doit décrire tous les plans à élaborer et à utiliser dans le cadre du projet.
- b. Les relations entre les plans de projet et une planification précise de leur préparation et de leur mise à jour doivent être décrites.

<6.3> Sûreté de fonctionnement et sécurité des logiciels

- a. Le SPAP doit comprendre une description et une justification des mesures à appliquer pour le traitement des logiciels critiques, notamment les analyses à réaliser et les standards applicables aux logiciels critiques.

<6.4> Gestion de la documentation et de la configuration logiciel

- a. Le SPAP doit décrire la contribution de la fonction d'assurance produit logiciel à la mise en œuvre appropriée de la gestion de la documentation et de la configuration.

- b. Le système de gestion des anomalies doit être décrits ou référencé. Le point du cycle de vie du logiciel à partir duquel s'appliquent les procédures relatives aux non-conformités doit être indiqué.
- c. Le SPAP doit identifier les méthodes et outils à utiliser pour protéger le logiciel fourni, un logiciel de calcul de clé de type checksum pour les logiciels opérationnels livrés et une méthode de marquage pour les supports fournis.

<6.5> Métriques sur les processus

- a. Le SPAP doit décrire les métriques sur les processus dérivées des modèles qualité définis, les méthodes de collecte, de conservation et d'analyse utilisées pour ces métriques et la façon dont ces dernières sont utilisées dans la gestion des processus de développement.

<6.6> Réutilisation de logiciels

- a. Le SPAP doit décrire l'approche de réutilisation de logiciels existants, et notamment le delta qualification.

<6.7> Planification de l'assurance produit pour les processus et activités individuels

- a. Les processus et activités suivants doivent être couverts, le domaine d'application et le cycle de vie du projet devant être pris en compte :
 1. analyse des exigences logiciel ;
 2. conception de l'architecture du logiciel et des éléments logiciel ;
 3. codage ;
 4. tests et validation (y compris les essais de non régression) ;
 5. vérification ;
 6. livraison et recette du logiciel ;
 7. exploitation et maintenance.

<6.8> Procédures et standards

- a. Le SPAP doit décrire ou référencer l'ensemble des procédures et des standards applicables au développement du logiciel dans le cadre du projet.
- b. Les mesures d'assurance produit logiciel prises afin de garantir le respect des procédures et standards du projet doivent être décrites.
- c. Les standards et procédures à décrire ou répertorier conformément au paragraphe B.2.1a doivent être au minimum celles traitant des aspects suivants :
 1. gestion de projet ;
 2. management des risques ;
 3. gestion de configuration et de documentation ;
 4. vérification et validation ;
 5. exigences d'ingénierie ;
 6. conception ;
 7. codage ;
 8. métriques ;
 9. traitement des anomalies ;
 10. audits ;
 11. alertes ;
 12. approvisionnement ;

13. réutilisation de logiciels existants ;
14. utilisation des méthodes et des outils ;
15. précision numérique ;
16. livraison, installation et recette ;
17. exploitation ;
18. maintenance ;
19. programmation et marquage des dispositifs.

<7> **Assurance qualité du produit logiciel**

- a. Le SPAP doit décrire l'approche adoptée pour garantir la qualité du produit logiciel.
- b. La description de l'approche définie dans le paragraphe B.2.1<7>a doit inclure les éléments suivants :
 1. spécification des métriques sur les produits, des valeurs cibles et des moyens utilisés pour les collecter ;
 2. définition d'un programme de métriques respectant les délais impartis ;
 3. analyses à réaliser sur les métriques collectées ;
 4. façon dont les résultats sont communiqués à l'équipe de développement ;
 5. exigences qualité relatives à la documentation ;
 6. activités d'assurance destinées à garantir que le produit est conforme aux exigences qualité.

<8> **Matrice de conformité aux exigences d'assurance produit logiciel**

- a. Le SPAP doit inclure (ou faire référence à) la matrice de conformité aux exigences d'assurance produit logiciel applicables (par exemple : paragraphes de l'ECSS-Q-ST-80 adaptés à l'aide d'un document d'exigences d'assurance produit).
- b. Pour chaque exigence d'assurance produit logiciel, les informations suivantes doivent être fournies :
 1. identification de l'exigence ;
 2. conformité
(C = conforme, NC = non conforme, NA = non applicable) ;
 3. référence à la documentation du projet relative à cette exigence (par exemple : section du plan d'assurance produit logiciel) ;
 4. remarques.

B.2.2 Remarques

Certaines parties du SPAP peuvent être combinée à celles du plan assurance produit du projet tel que défini dans l'ECSS-Q-ST-10.

Annexe C (normative)

Bilans qualité logiciel de jalon (SPAMR) - DRD

C.1 Identification de la DRD

C.1.1 Identification des exigences et document source

Le bilan qualité logiciel de jalon (SPAMR) est demandé par les exigences normatives résumées dans le Tableau C-1.

Tableau C-1 : Traçabilité du SPAP par rapport aux paragraphes des standards ECSS-E-ST-40 et ECSS-Q-ST-80

Standard ECSS	Paragraphe	Section de la DRD
ECSS-Q-ST-80	5.2.2.3	Toutes
	5.6.1.2	<5>.a
	5.6.1.3	<5>.b
	6.2.5.4	<7>
	6.2.5.5	<7>
	6.2.6.3	<4>
	6.2.6.7	<4>
	6.2.8.5	<6>
	6.3.3.4	<6>
	6.3.3.6	<6>.a
	6.3.3.7	<6>.b
	6.3.4.7	<6>
	6.3.5.3	<8>
	6.3.5.5	<8>
	6.3.5.12	<8>
	7.1.6	<7>
	7.1.8	<7>

C.1.2 Objet

Le Bilan qualité logiciel de jalon fait partie du dossier d'assurance produit (PAF).

Le bilan qualité logiciel de jalon vise principalement à réunir et à présenter, au niveau des jalons du projet, les rapports associés aux activités d'assurance produit logiciel réalisées au cours des phases précédentes du projet.

C.2 Réponse attendue

C.2.1 Domaine d'application et contenu

<1> Introduction

- a. Le SPAMR doit contenir une description du but, de l'objectif, du contenu et de la raison de son élaboration.

<2> Documents applicables et de référence

- a. Le SPAMR doit répertorier les documents applicables et de référence utilisés en support de sa rédaction.

<3> Termes, définitions et abréviations

- a. Le SPAMR doit inclure tous les autres termes, définitions et abréviations utilisés.

<4> Activités de vérification effectuées

- a. Le SPAMR doit contenir des rapports sur les activités de vérification effectuées par la fonction d'assurance produit, et notamment :
 1. les revues ;
 2. les contrôles ;
 3. les contrôles transverses ;
 4. la revue des matrices de traçabilité ;
 5. les documents passés en revue.
- b. Le SPAMR doit contenir des rapports sur la vérification des mesures appliquées pour le traitement des logiciels critiques.

<5> Méthodes et outils

- a. Le SPAMR doit inclure ou référencer une justification de l'adéquation des méthodes et des outils appliqués dans toutes les activités du cycle de développement, notamment l'analyse des exigences, la spécification du logiciel, la conception, le codage, la validation, les tests, la gestion de configuration, la vérification et l'assurance produit.
- b. Le SPAMR doit inclure des rapports sur l'utilisation correcte des méthodes et des outils.

<6> Respect des standards de conception et de codage

- a. Le SPAMR doit comprendre des rapports sur la conformité des produits logiciel aux standards de modélisation, de conception et de codage applicables, notamment :
 - 1. un rapport sur l'application de mesures destinées à garantir que la complexité et la modularité de la conception répondent aux exigences qualité ;
 - 2. un rapport sur la documentation de conception par rapport à la conformité pour la maintenance.

<7> Métriques sur les produits et les processus

- a. Le SPAMR doit inclure des rapports sur les métriques sur les produits et les processus collectées, les analyses correspondantes effectuées, les actions correctives entreprises et l'état de ces actions.
- b. Les résultats de l'analyse de la maturité du logiciel doivent également être consignés.

<8> Tests et validation

- a. Le SPAMR doit inclure des rapports sur l'adéquation de la documentation relative aux tests et à la validation (notamment la faisabilité, la traçabilité et la reproductibilité) et sur la couverture des tests par rapport aux objectifs établis.

<9> Problèmes et anomalies logiciel

- a. Le SPAMR doit inclure des rapports sur l'état des problèmes et des non-conformités associées au logiciel.

<10> Références aux rapports d'avancement

- a. Lorsque des informations pertinentes et à jour ont déjà été communiquées dans le cadre des rapports d'avancement AP régulièrement fournis, un résumé représentatif doit être présenté, avec des références détaillées au(x) rapport(s) d'avancement contenant ces informations.

C.2.2 Remarques

Certaines parties de ce document peuvent être combinées à celles associées au rapport d'assurance produit du projet tel que défini dans l'ECSS-Q-ST-10.

Annexe D (normative)

Adaptation du standard en fonction de la criticité des logiciels

D.1 Catégories de criticité des logiciels

Les classes de criticité logiciel suivantes sont définies en fonction de la gravité des conséquences des défaillances système (voir le Tableau 5-1 de l'ECSS-Q-ST-40 et le Tableau 6-1 de l'ECSS-Q-ST-30).

Tableau D-1 : Catégories de criticité des logiciels

Catégorie	Définition
A	Logiciels dont la non-exécution, l'exécution incorrecte ou le comportement anormal peuvent provoquer (ou contribuer à) l'apparition d'une défaillance du système à l'origine de : → Conséquences désastreuses
B	Logiciels dont la non-exécution, l'exécution incorrecte ou le comportement anormal peuvent provoquer (ou contribuer à) l'apparition d'une défaillance du système à l'origine de : → Conséquences critiques
C	Logiciels dont la non-exécution, l'exécution incorrecte ou le comportement anormal peuvent provoquer (ou contribuer à) l'apparition d'une défaillance du système à l'origine de : → Conséquences majeures
D	Logiciels dont la non-exécution, l'exécution incorrecte ou le comportement anormal peuvent provoquer (ou contribuer à) l'apparition d'une défaillance du système à l'origine de : → Conséquences mineures ou négligeables

D.2 Matrice d'applicabilité

La matrice d'applicabilité suivante représente une adaptation des exigences du présent standard en fonction des catégories de criticité des logiciels définies dans le paragraphe D.1.

Pour chaque paragraphe du présent standard et chaque catégorie de criticité, il est indiqué si le paragraphe est applicable (O), non applicable (N) ou applicable dans les conditions spécifiques décrites ci-après.

Tableau D-2 : Matrice d'applicabilité en fonction de la criticité des logiciels

Para- gra- phe	Description	A	B	C	D
5	Mise en œuvre du programme d'assurance produit logiciel	-	-	-	-
5.1	Organisation et responsabilités	-	-	-	-
5.1.1	Organisation	O	O	O	O
5.1.2	Responsabilités et autorité	-	-	-	-
5.1.2.1		O	O	O	O
5.1.2.2		O	O	O	O
5.1.2.3		O	O	O	O
5.1.3	Ressources	-	-	-	-
5.1.3.1		O	O	O	O
5.1.3.2		O	O	O	N
5.1.4	Responsable/ingénieur assurance produit logiciel	-	-	-	-
5.1.4.1		O	O	O	O
5.1.4.2		O	O	O	O
5.1.5	Formation	-	-	-	-
5.1.5.1		O	O	O	Résultats attendus non exigés
5.1.5.2		O	O	O	N
5.1.5.3		O	O	O	O
5.1.5.4		O	O	O	O
5.2	Gestion du programme d'assurance produit logiciel	-	-	-	-
5.2.1	Planification et contrôle de l'assurance produit logiciel	-	-	-	-
5.2.1.1		O	O	O	O
5.2.1.2		O	O	O	O
5.2.1.3		O	O	O	O
5.2.1.4		O	O	O	O
5.2.1.5		O	O	O	O

Para- gra- phe	Description	A	B	C	D
5.2.2	Bilans qualité logiciel	-	-	-	-
5.2.2.1		O	O	O	O
5.2.2.2		O	O	O	O
5.2.2.3		O	O	O	O
5.2.3	Audits	O	O	O	Audits planifiés et réalisés Uniquement si nécessaire
5.2.4	Alertes	O	O	O	O
5.2.5	Problèmes logiciel	-	-	-	-
5.2.5.1		O	O	O	O
5.2.5.2		O	O	O	O
5.2.5.3		O	O	O	O
5.2.5.4		O	O	O	O
5.2.6	Non-conformités	-	-	-	-
5.2.6.1		O	O	O	O
5.2.6.2		O	O	O	O
5.2.7	Exigences et modèles qualité	-	-	-	-
5.2.7.1		O	O	O	O
5.2.7.2		O	O	O	Caractéristiques pertinentes seulement (par ex., la sécurité ne s'applique pas aux logiciels de cat. D)
5.3	Management des risques et maîtrise des éléments critiques	-	-	-	-
5.3.1	Management des risques	O	O	O	O
5.3.2	Maîtrise des éléments critiques	-	-	-	-
5.3.2.1		O	O	O	O
5.3.2.2		O	O	O	O
5.4	Sélection et suivi des fournisseurs	-	-	-	-
5.4.1	Sélection des fournisseurs	O	O	O	O
5.4.1.1		O	O	O	Résultats attendus non exigés
5.4.1.2		O	O	O	O
5.4.2	Exigences pour les fournisseurs	-	-	-	-
5.4.2.1		O	O	O	O

Para- gra- phe	Description	A	B	C	D
5.4.2.2		O	O	O	N
5.4.3	Surveillance des fournisseurs	-	-	-	-
5.4.3.1		O	O	O	O
5.4.3.2		O	O	O	O
5.4.3.3		O	O	O	O
5.4.3.4		O	O	O	N
5.4.4	Classification de la criticité	O	O	O	O
5.5	Approvisionnement	-	-	-	-
5.5.1	Documents d'approvisionnement	O	O	O	O
5.5.2	Revue de la liste des composants logiciel acquis	O	O	O	O
5.5.3	Caractérisation des approvisionnements	O	O	O	O
5.5.4	Identification	O	O	O	O
5.5.5	Contrôle	O	O	O	O
5.5.6	Export	O	O	O	O
5.6	Outils et environnement support	-	-	-	-
5.6.1	Méthodes et outils	-	-	-	-
5.6.1.1		O	O	O	Les méthodes et outils proposés doivent avoir été utilisés avec succès dans au moins un projet antérieur (éventuellement non spatial)
5.6.1.2		O	O	O	Résultats attendus non exigés
5.6.1.3		O	O	O	Résultats attendus non exigés
5.6.2	Sélection de l'environnement de développement	-	-	-	-
5.6.2.1		O	O	O	Résultats attendus non exigés
5.6.2.2		O	O	O	Résultats attendus non exigés
5.6.2.3		O	O	O	O

Para- gra- phe	Description	A	B	C	D
5.7	Processus d'évaluation et d'amélioration	-	-	-	-
5.7.1	Evaluation des processus	O	O	O	N
5.7.2	Processus d'évaluation	-	-	-	-
5.7.2.1		O	O	O	N
5.7.2.2		O	O	O	N
5.7.2.3		O	O	O	N
5.7.2.4		O	O	O	N
5.7.3	Amélioration des processus	-	-	-	-
5.7.3.1		O	O	O	N
5.7.3.2		O	O	O	N
5.7.3.3		O	O	O	N
6	Assurance des processus logiciel	-	-	-	-
6.1	Cycle de vie du développement logiciel	-	-	-	-
6.1.1	Définition du cycle de vie	O	O	O	O
6.1.2	Objectifs de qualité	O	O	O	O
6.1.3	Revue de la définition du cycle de vie	O	O	O	O
6.1.4	Ressources du cycle de vie	O	O	O	O
6.1.5	Planification du processus de validation logiciel	O	O	O	O
6.2	Exigences applicables à tous les processus d'ingénierie logiciel	-	-	-	-
6.2.1	Documentation des processus	-	-	-	-
6.2.1.1		O	O	O	O
6.2.1.2		O	O	O	O
6.2.1.3		O	O	O	O
6.2.1.4		O	O	O	O
6.2.1.5		O	O	O	O
6.2.1.6		O	O	O	O
6.2.1.7		O	O	O	O
6.2.1.8		O	O	O	O
6.2.1.9		O	O	O	N
6.2.2	Sûreté de fonctionnement et sécurité des logiciels	-	-	-	-
6.2.2.1		O	O	O	O
6.2.2.2		O	O	O	N
6.2.2.3		O	O	O	N
6.2.2.4		O	O	O	N
6.2.2.5		O	O	O	N

Para- gra- phe	Description	A	B	C	D
6.2.2.6		O	O	O	N
6.2.2.7		O	O	O	N
6.2.2.8		O	O	O	O
6.2.2.9		O	O	O	O
6.2.3	Traitement des logiciels critiques	-	-	-	-
6.2.3.1		O	O	O	O
6.2.3.2		O	O	O	N
6.2.3.3		O	O	O	N
6.2.3.4		O	O	O	N
6.2.3.5		O	O	O	N
6.2.3.6		O	O	O	N
6.2.3.7		O	O	O	N
6.2.3.8		O	O	N	N
6.2.4	Gestion de configuration des logiciels	-	-	-	-
6.2.4.1		O	O	O	O
6.2.4.2		O	O	O	O
6.2.4.3		O	O	O	O
6.2.4.4		O	O	O	O
6.2.4.5		O	O	O	O
6.2.4.6		O	O	O	O
6.2.4.7		O	O	O	O
6.2.4.8		O	O	O	O
6.2.4.9		O	O	O	O
6.2.4.10		O	O	O	O
6.2.4.11		O	O	O	O
6.2.5	Métriques sur les processus	-	-	-	-
6.2.5.1		O	O	O	O
6.2.5.2		O	O	O	O
6.2.5.3		O	O	O	O
6.2.5.4		O	O	O	Limité au nombre de problèmes détectés pendant la validation
6.2.5.5		O	O	O	O
6.2.6	Vérification	-	-	-	-
6.2.6.1		O	O	O	O
6.2.6.2		O	O	O	O
6.2.6.3		O	O	O	O

Para- gra- phe	Description	A	B	C	D
6.2.6.4		O	O	O	O
6.2.6.5		O	O	O	N
6.2.6.6		O	O	O	N
6.2.6.7		O	O	O	O
6.2.6.8		O	O	O	O
6.2.6.9		O	O	O	O
6.2.6.10		O	O	O	O
6.2.6.11		O	O	O	O
6.2.6.12		O	O	O	O
6.2.6.13		O	O	N	N
6.2.7	Réutilisation de logiciels existants	-	-	-	-
6.2.7.1		O	O	O	O
6.2.7.2		O	O	O	O
6.2.7.3		O	O	O	O
6.2.7.4		O	O	O	Points 3, 4, 5 et 7 non applicables Point 2 limité à la conception de l'architecture
6.2.7.5		O	O	O	O
6.2.7.6		O	O	O	O
6.2.7.7		O	O	O	Limité jusqu'à la garantie de la maintenabilité du logiciel
6.2.7.8		O	O	O	Limité jusqu'à la garantie de la maintenabilité du logiciel
6.2.7.9		O	O	O	O
6.2.7.10		O	O	O	O
6.2.7.11		O	O	O	O
6.2.8	Génération automatique de codes	-	-	-	-
6.2.8.1		O	O	O	O
6.2.8.2		O	O	O	O
6.2.8.3		O	O	O	O
6.2.8.4		O	O	O	O
6.2.8.5		O	O	O	O
6.2.8.6		O	O	O	O

Para- gra- phe	Description	A	B	C	D
6.2.8.7		O	O	O	O
6.3	Exigences applicables à chacun des processus ou activités d'ingénierie logiciel	-	-	-	-
6.3.1	Traitement des exigences système relatives au logiciel	-	-	-	-
6.3.1.1		O	O	O	O
6.3.1.2		O	O	O	O
6.3.1.3		O	O	O	O
6.3.2	Analyse des exigences logiciel	-	-	-	-
6.3.2.1		O	O	O	O
6.3.2.2		O	O	O	O
6.3.2.3		O	O	O	O
6.3.2.4		O	O	O	O
6.3.2.5		O	O	O	O
6.3.3	Conception de l'architecture du logiciel et des éléments logiciel	-	-	-	-
6.3.3.1		O	O	O	Contrôle de la documentation uniquement
6.3.3.2		O	O	O	Seulement recommandé
6.3.3.3		O	O	O	N
6.3.3.4		O	O	O	Seulement si les standards de conception sont appliqués (6.3.2.2)
6.3.3.5		O	O	O	N
6.3.3.6		O	O	O	N
6.3.3.7		O	O	O	O
6.3.4	Codage	-	-	-	-
6.3.4.1		O	O	O	O
6.3.4.2		O	O	O	O
6.3.4.3		O	O	O	N
6.3.4.4		O	O	O	N
6.3.4.5		O	O	O	O
6.3.4.6		O	O	O	O
6.3.4.7		O	O	O	O

Para- gra- phe	Description	A	B	C	D
6.3.4.8		O	O	O	Le code doit être placé sous contrôle de configuration au début des essais de validation
6.3.5	Essais et validation	-	-	-	-
6.3.5.1		O	O	O	Aucune activité de test unitaires et d'intégration formelle requise
6.3.5.2		O	O	O	Aucune activité de tests unitaires et d'intégration formelle requise
6.3.5.3		O	O	O	Procédures et données de test vérifiées par échantillon
6.3.5.4		O	O	O	Applicable aux tests de validation et de recette uniquement
6.3.5.5		O	O	O	O
6.3.5.6		O	O	O	O
6.3.5.7		O	O	O	O
6.3.5.8		O	O	O	O
6.3.5.9		O	O	O	N
6.3.5.10		O	O	O	N
6.3.5.11		O	O	O	O
6.3.5.12		O	O	O	O
6.3.5.13		O	O	O	O
6.3.5.14		O	O	O	Applicable aux tests de validation et de recette uniquement
6.3.5.15		O	O	O	O
6.3.5.16		O	O	O	O
6.3.5.17		O	O	O	O
6.3.5.18		O	O	O	O
6.3.5.19		O	O	O	N
6.3.5.20		O	O	O	O

Para- gra- phe	Description	A	B	C	D
6.3.5.21		O	O	O	O
6.3.5.22		O	O	O	O
6.3.5.23		O	O	O	O
6.3.5.24		O	O	O	O
6.3.5.25		O	O	O	O
6.3.5.26		O	O	O	O
6.3.5.27		O	O	O	O
6.3.5.28		O	O	N	N
6.3.5.29		O	O	O	O
6.3.5.30		O	O	O	N
6.3.5.31		O	O	O	N
6.3.5.32		O	O	O	O
6.3.6	Livraison et recette des logiciels	-	-	-	-
6.3.6.1		O	O	O	O
6.3.6.2		O	O	O	O
6.3.6.3		O	O	O	O
6.3.6.4		O	O	O	O
6.3.6.5		O	O	O	O
6.3.6.6		O	O	O	O
6.3.6.7		O	O	O	O
6.3.6.8		O	O	O	O
6.3.6.9		O	O	O	O
6.3.7	Opérations	-	-	-	-
6.3.7.1		O	O	O	O
6.3.7.2		O	O	Point sur les fonctionnalités de sécurité non applicable	Point sur les fonctionnalités de sécurité non applicable
6.3.7.3		O	O	O	O
6.3.8	Maintenance	-	-	-	-
6.3.8.1		O	O	O	O
6.3.8.2		O	O	O	O
6.3.8.3		O	O	O	O
6.3.8.4		O	O	O	O
6.3.8.5		O	O	O	O

Para- gra- phe	Description	A	B	C	D
6.3.8.6		O	O	O	O
6.3.8.7		O	O	O	Données statistiques non collectées
7	Assurance qualité du produit logiciel	-	-	-	-
7.1	Objectifs de qualité produit et métriques	-	-	-	-
7.1.1	Déclinaison des exigences	O	O	O	O
7.1.2	Définition quantitative des exigences qualité	O	O	O	O
7.1.3	Activités d'assurance relatives aux exigences qualité produit	O	O	O	O
7.1.4	Métriques sur les produits	O	O	O	Point 4(a) non applicable
7.1.5	Métriques de base	O	O	O	Métriques relatives à la conception, au taux de fautes et au taux de défaillances non exigées
7.1.6	Rapports sur les métriques	O	O	O	O
7.1.7	Précision numérique	O	O	O	N
7.1.8	Analyse de la maturité du logiciel	O	O	O	N
7.2	Exigences qualité produit	-	-	-	-
7.2.1	Exigences de besoin et spécification technique	-	-	-	-
7.2.1.1		O	O	O	O
7.2.1.2		O	O	O	O
7.2.1.3		O	O	O	O
7.2.2	Conception et documentation associée	-	-	-	-
7.2.2.1		O	O	O	O
7.2.2.2		O	O	O	O
7.2.2.3		O	O	O	O
7.2.3	Documentation de test et de validation	-	-	-	-
7.2.3.1		O	O	O	O
7.2.3.2		O	O	O	O
7.2.3.3		O	O	O	O
7.2.3.4		O	O	O	O
7.2.3.5		O	O	O	O
7.2.3.6		O	O	O	O

Para- gra- phe	Description	A	B	C	D
7.3	Logiciels destinés à être réutilisés	-	-	-	-
7.3.1	Exigences du client	O	O	O	O
7.3.2	Documentation séparée	O	O	O	O
7.3.3	Informations autosuffisantes	O	O	O	O
7.3.4	Exigences relatives aux réutilisations prévues	O	O	O	O
7.3.5	Gestion de configuration en vue d'une réutilisation	O	O	O	O
7.3.6	Essais sur plusieurs plates-formes	O	O	O	O
7.3.7	Certificat de conformité	O	O	O	O
7.4	Matériel standard pour un système opérationnel	-	-	-	-
7.4.1	Acquisition de matériel	O	O	O	O
7.4.2	Acquisition de services	O	O	O	O
7.4.3	Contraintes	O	O	O	O
7.4.4	Sélection	O	O	O	O
7.4.5	Maintenance	O	O	O	O
7.5	Microprogrammes	-	-	-	-
7.5.1	Programmation des dispositifs	O	O	O	O
7.5.2	Marquage	O	O	O	O
7.5.3	Etalonnage	O	O	O	O

Annexe E (informative)

Liste des exigences adaptables

Les exigences suivantes s'appliquent dans des conditions précises, comme indiqué dans le texte correspondant.

5.1.4.2	Le <i>responsable ingénieur</i> assurance produit logiciel doit rendre des comptes au chef de projet (via le responsable assurance produit du projet, <i>le cas échéant</i>).
5.2.2.1	Le fournisseur doit régulièrement rendre compte de l'avancement de la mise en œuvre du programme d'assurance produit logiciel, <i>le cas échéant</i> dans le cadre du rapport d'assurance produit général du projet.
6.2.3.4	<i>Dans le cas</i> de modifications mineures sur les outils ayant une incidence sur la génération du code exécutable, une comparaison binaire du code exécutable généré par les différents outils peut être utilisée pour vérifier qu'aucun changement n'a été introduit.
6.2.6.13	Cette exigence n'est applicable que <i>lorsque</i> les risques associés au projet justifient les coûts induits. Le client <i>peut</i> envisager un niveau d'indépendance moindre, par exemple une équipe indépendante au sein de la même entité.

Les exigences suivantes prévoient un accord entre le client et le fournisseur.

6.3.2.5	Préalablement à l'élaboration de la spécification technique, le client et le fournisseur doivent au minimum se mettre d'accord sur les principes et règles suivants : [...].
6.3.5.2	En fonction de la criticité des logiciels, les objectifs de couverture d'essai pour chaque niveau d'essai doivent être convenus entre le client et le fournisseur. Leur réalisation doit être contrôlée par des métriques associées : [...].

Annexe F (informative)

Organisation et contenu des documents à chaque jalon

F.1 Introduction

Le tableau suivant montre l'organisation des Résultats attendus des paragraphes du présent standard, triés par revue, puis par dossier cible et par DRD.

Lorsque aucune DRD n'est disponible, le symbole « - » est indiqué.

F.2 Résultats attendus de l'ECSS-Q-ST-80 - SRR

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
7.1.1.a	Exigences de besoin	RB	SSS	<5.9>
7.1.2.a	Exigences de besoin	RB	SSS	<5.9>
7.2.1.1.a	Exigences de besoin	RB	SSS	<5.9>
7.2.1.3.a	Exigences de besoin	RB	SSS	<5.1>
5.4.4	Résultats des analyses de sécurité et de sûreté de fonctionnement des fournisseurs de niveau inférieur	RB	-	
5.1.2.1	Plan assurance produit logiciel	PAF	SPAP	<5.1>
5.1.2.2	Plan assurance produit logiciel	PAF	SPAP	<5.1>, <5.2>
5.1.2.3	Plan assurance produit logiciel	PAF	SPAP	<5.1>
5.1.3.1	Plan assurance produit logiciel	PAF	SPAP	<5.3>
5.1.3.2	Plan assurance produit logiciel	PAF	SPAP	<5.1>, <5.3>
5.1.4.1	Plan assurance produit logiciel	PAF	SPAP	<5.1>, <5.3>
5.2.1.1	Plan assurance produit logiciel	PAF	SPAP	Toutes
5.2.1.5	Plan assurance produit logiciel	PAF	SPAP	<8>
5.2.6.1.a.a	Procédure de gestion des anomalies logiciel du plan assurance produit logiciel	PAF	SPAP	
5.2.6.2.	Plan assurance produit logiciel	PAF	SPAP	<6.4>
5.6.1.1	Plan assurance produit logiciel	PAF	SPAP	<5.8>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.1.1	Plan assurance produit logiciel	PAF	SPAP	<6.1>
6.1.5	Plan assurance produit logiciel	PAF	SPAP	<6.1>
6.2.1.4	Plan assurance produit logiciel	PAF	SPAP	<6.2>
6.2.4.8.a	Plan assurance produit logiciel	PAF	SPAP	<6.4>
6.2.4.9	Plan assurance produit logiciel	PAF	SPAP	<6.4>
6.2.4.11.a	Plan assurance produit logiciel	PAF	SPAP	<6.4>
6.2.5.1	Plan assurance produit logiciel	PAF	SPAP	<6.5>
6.2.5.2	Plan assurance produit logiciel	PAF	SPAP	<6.5>
6.2.7.2.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.2.7.3.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.2.7.4.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.2.7.5.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
7.1.3	Plan assurance produit logiciel	PAF	SPAP	<7>
7.1.4	Plan assurance produit logiciel	PAF	SPAP	<7>
7.1.5	Plan assurance produit logiciel	PAF	SPAP	<7>
7.2.2.3.a	Plan assurance produit logiciel	PAF	SPAP	<6.7>
5.2.2.3	Bilans qualité logiciel de jalon	PAF	SPAMR	Toutes
5.6.1.2	Bilans qualité logiciel de jalon	PAF	SPAMR	<5>
6.2.6.12	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
5.2.3	Plan et calendrier des audits	PAF	-	
5.4.2.1	Exigences d'assurance produit logiciel applicables aux fournisseurs	PAF	-	
5.4.2.2	Exigences d'assurance produit logiciel applicables aux fournisseurs	PAF	-	
6.2.2.1	Classification de la criticité des produits logiciel	PAF	-	
6.2.8.4	Standards de modélisation	PAF	-	
6.3.3.2	Standards de conception	PAF	-	
7.4.1.b	Compte rendu des contrôles à la réception	PAF	-	
5.5.2	Plan de développement logiciel	MGT	SDP	<4.8>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
5.6.2.1	Plan de développement logiciel	MGT	SDP	<5.4>
5.6.2.2	Plan de développement logiciel	MGT	SDP	<5.4>
6.2.4.2	Plan de gestion de configuration des logiciels	MGT	SCMP	
7.3.5	Gestion de configuration des composants réutilisables	MGT	SCMP	
5.1.5.1	Plan de formation	MGT	-	
5.2.6.1.b	Identification des experts logiciel dans la commission de traitement des anomalies logiciel	MGT	-	
5.5.3	Données d'approvisionnement	MGT	-	
6.2.7.2.b	Dossier des logiciels réutilisés	DJF	SRF	<6>
6.2.7.3.b	Dossier des logiciels réutilisés	DJF	SRF	<4>, <5>
6.2.7.4.b	Dossier des logiciels réutilisés	DJF	SRF	<5>
6.2.7.5.b	Dossier des logiciels réutilisés	DJF	SRF	<6>
6.2.7.6	Dossier des logiciels réutilisés	DJF	SRF	<4>, <5>
6.2.7.7	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.8	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.11	Dossier des logiciels réutilisés	DJF	SRF	<9>
6.2.6.4	Fiches d'anomalies logicielles	DJF	-	
6.2.6.13.a	Plan ISVV	DJF	-	
6.3.5.8	Problèmes logiciel	DJF	-	
6.3.5.28.a	Plan ISVV	DJF	-	
7.4.1.a	Justification de la sélection des équipements au sol opérationnels	DJF	-	
7.4.2	Justification de la sélection des services de support opérationnels	DJF	-	
7.4.3	Justification de la sélection des équipements au sol opérationnels	DJF	-	
7.4.4	Justification de la sélection des équipements au sol opérationnels	DJF	-	

F.3 Résultats attendus de l'ECSS-Q-ST-80 - PDR

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.3.2.4	Spécification des exigences logiciel	ST	SRS	<5>
7.1.1.b	Spécification technique	ST	SRS	<5.10>
7.1.2.b	Spécification technique	ST	SRS	<5.10>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
7.2.1.1.b	Spécification technique	ST	SRS	<5.10>
7.2.1.3.b	Spécification technique	ST	SRS	<6>
7.3.4	Spécification technique des composants réutilisables	ST	-	
5.2.1.1	Plan assurance produit logiciel	PAF	SPAP	Toutes
5.2.1.5	Plan assurance produit logiciel	PAF	SPAP	<8>
5.2.6.2.	Plan assurance produit logiciel	PAF	SPAP	<6.4>
5.2.7.1	Plan assurance produit logiciel	PAF	SPAP	<5.5>
5.2.7.2	Plan assurance produit logiciel	PAF	SPAP	<5.5>
5.4.3.3	Plan assurance produit logiciel des fournisseurs de niveau immédiatement inférieur	PAF	SPAP	Toutes
5.4.3.4	Plan assurance produit logiciel des fournisseurs de niveau immédiatement inférieur	PAF	SPAP	Toutes
5.6.1.1	Plan assurance produit logiciel	PAF	SPAP	<5.8>
6.1.1	Plan assurance produit logiciel	PAF	SPAP	<6.1>
6.1.5	Plan assurance produit logiciel	PAF	SPAP	<6.1>
6.2.1.4	Plan assurance produit logiciel	PAF	SPAP	<6.2>
6.2.3.1.a	Plan assurance produit logiciel	PAF	SPAP	<6.3>
6.2.3.2	Plan assurance produit logiciel	PAF	SPAP	<6.3>
6.2.3.4	Plan assurance produit logiciel	PAF	SPAP	<6.7>
6.2.3.5	Plan assurance produit logiciel	PAF	SPAP	<6.7>
6.2.4.8.a	Plan assurance produit logiciel	PAF	SPAP	<6.4>
6.2.4.9	Plan assurance produit logiciel	PAF	SPAP	<6.4>
6.2.4.11.a	Plan assurance produit logiciel	PAF	SPAP	<6.4>
6.2.5.1	Plan assurance produit logiciel	PAF	SPAP	<6.5>
6.2.5.2	Plan assurance produit logiciel	PAF	SPAP	<6.5>
6.2.7.2.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.2.7.3.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.2.7.4.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.2.7.5.a	Approche de réutilisation des logiciels, notamment l'approche de delta qualification	PAF	SPAP	<6.6>
6.3.3.3	Plan assurance produit logiciel	PAF	SPAP	<6.8>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.3.3.5	Plan assurance produit logiciel	PAF	SPAP	<6.7>
6.3.3.7.a	Plan assurance produit logiciel	PAF	SPAP	<6.7>
6.3.4.3	Plan assurance produit logiciel	PAF	SPAP	<6.8>
6.3.4.6	Plan assurance produit logiciel	PAF	SPAP	<6.8>
6.3.5.1	Plan assurance produit logiciel	PAF	SPAP	<6.7>
6.3.5.2	Plan assurance produit logiciel	PAF	SPAP	<6.7>
7.1.3	Plan assurance produit logiciel	PAF	SPAP	<7>
7.1.4	Plan assurance produit logiciel	PAF	SPAP	<7>
7.1.5	Plan assurance produit logiciel	PAF	SPAP	<7>
7.2.2.3.a	Plan assurance produit logiciel	PAF	SPAP	<6.7>
7.5.1	Plan assurance produit logiciel	PAF	SPAP	<6.8>
7.5.2	Plan assurance produit logiciel	PAF	SPAP	<6.8>
5.2.2.3	Bilans qualité logiciel de jalon	PAF	SPAMR	Toutes
5.6.1.2	Bilans qualité logiciel de jalon	PAF	SPAMR	<5>
6.2.3.3	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
6.2.6.12	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
5.2.5.1	Procédures de traitement des problèmes logiciel	PAF	-	
5.2.5.2	Procédures de traitement des problèmes logiciel	PAF	-	
5.2.5.3	Procédures de traitement des problèmes logiciel	PAF	-	
5.5.5	Compte rendu des contrôles à la réception	PAF	-	
6.2.1.6	Procédures et standards	PAF	-	
6.2.1.7	Procédures et standards	PAF	-	
6.2.2.1	Classification de la criticité des produits logiciel	PAF	-	
6.2.2.2	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.2.3	Classification de la criticité des composants logiciel	PAF	-	
6.2.2.7	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.3.1.b	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.8.4	Standards de modélisation	PAF	-	

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.3.3.2	Standards de conception	PAF	-	
6.3.4.1	Standards de codage	PAF	-	
6.3.4.2	Standards de codage	PAF	-	
6.3.4.4	Standards de codage et description des outils	PAF	-	
7.4.1.b	Compte rendu des contrôles à la réception	PAF	-	
5.5.2	Plan de développement logiciel	MGT	SDP	<4.8>
5.6.2.1	Plan de développement logiciel	MGT	SDP	<5.4>
5.6.2.2	Plan de développement logiciel	MGT	SDP	<5.4>
6.3.4.5	Plan de développement logiciel	MGT	SDP	<5.4>
6.2.4.2	Plan de gestion de configuration des logiciels	MGT	SCMP	
7.3.5	Gestion de configuration des composants réutilisables	MGT	SCMP	
5.5.3	Données d'approvisionnement	MGT	-	
7.1.7	Analyse de la précision numérique	DJF	SVR	<6>
6.2.6.1	Plan de vérification logiciel	DJF	SVerP	<6.3>
6.2.8.2	Documentation de validation et de test	DJF	SValP	<4.1>
6.2.8.7	Documentation de validation et de test	DJF	SValP	<4.1>
6.3.5.22	Documentation de test et de validation	DJF	SValP	<4>
6.3.5.23	Documentation de test et de validation	DJF	SValP	<4.4>
6.3.5.24	Documentation de test et de validation	DJF	SValP	<4.6>
6.3.5.25	Documentation de test et de validation	DJF	SValP	<5>
6.3.5.29	Documentation de test et de validation	DJF	SValP	<6>
6.2.8.2	Documentation de validation et de test	DJF	SUITP	<7.6>
6.2.8.7	Documentation de validation et de test	DJF	SUITP	<7.6>
6.3.5.22	Documentation de test et de validation	DJF	SUITP	<5>
6.3.5.23	Documentation de test et de validation	DJF	SUITP	<5.3>
6.3.5.24	Documentation de test et de	DJF	SUITP	<5.5>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
	validation			
6.3.5.25	Documentation de test et de validation	DJF	SUITP	<9.2>, <10>
6.2.7.2.b	Dossier des logiciels réutilisés	DJF	SRF	<6>
6.2.7.3.b	Dossier des logiciels réutilisés	DJF	SRF	<4>, <5>
6.2.7.4.b	Dossier des logiciels réutilisés	DJF	SRF	<5>
6.2.7.5.b	Dossier des logiciels réutilisés	DJF	SRF	<6>
6.2.7.6	Dossier des logiciels réutilisés	DJF	SRF	<4>, <5>
6.2.7.7	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.8	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.11	Dossier des logiciels réutilisés	DJF	SRF	<9>
6.2.6.4	Fiches d'anomalies logicielles	DJF	-	
6.2.6.13.a	Plan ISVV	DJF	-	
6.2.6.13.a	Rapport ISVV	DJF	-	
6.3.5.8	Fiches d'anomalies logicielles	DJF	-	
6.3.5.28.a	Plan ISVV	DJF	-	
6.3.5.28.b	Rapport ISVV	DJF	-	
7.4.1.a	Justification de la sélection des équipements au sol opérationnels	DJF	-	
7.4.2	Justification de la sélection des services de support opérationnels	DJF	-	
7.4.3	Justification de la sélection des équipements au sol opérationnels	DJF	-	
7.4.4	Justification de la sélection des équipements au sol opérationnels	DJF	-	
7.2.2.3.b	Justification des choix de conception	DD	SDD	<4.5>

F.4 Résultats attendus de l'ECSS-Q-ST-80 - CDR

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
5.2.1.3	Plan assurance produit logiciel	PAF	SPAP	Toutes
6.2.3.1.a	Plan assurance produit logiciel	PAF	SPAP	<6.3>
6.2.3.2	Plan assurance produit logiciel	PAF	SPAP	<6.3>
6.2.3.4	Plan assurance produit logiciel	PAF	SPAP	<6.7>
6.2.3.5	Plan assurance produit logiciel	PAF	SPAP	<6.7>
5.2.2.3	Bilans qualité logiciel de jalon	PAF	SPAMR	Toutes
6.2.3.3	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.2.6.12	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
5.5.5	Compte rendu des contrôles à la réception	PAF	-	
6.2.2.5	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.2.6	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.2.7	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.3.1.b	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.3.5.7	Déclaration de conformité aux plans et aux procédures de test	PAF	-	
6.3.5.11	Déclaration de conformité aux plans et aux procédures de test	PAF	-	
6.2.8.2	Documentation de validation et de test	DJF	SVS	<5.6>
6.2.8.7	Documentation de validation et de test	DJF	SVS	<5.6>
6.3.5.25	Documentation de test et de validation	DJF	SVS	<7.2>, <8>
6.3.5.29	Documentation de test et de validation	DJF	SVS	<6>
6.3.5.32	Spécification de validation du logiciel	DJF	SVS	<5>
6.2.6.5	Rapport de vérification logiciel	DJF	SVR	<4.4>
6.2.6.6	Rapport de vérification logiciel	DJF	SVR	<4.4>
7.1.7	Analyse de la précision numérique	DJF	SVR	<6>
7.2.3.6	Rapport de vérification logiciel	DJF	SVR	<4.5>
6.2.8.2	Documentation de validation et de test	DJF	SUITP	<7.6>
6.2.8.7	Documentation de validation et de test	DJF	SUITP	<7.6>
6.3.5.22	Documentation de test et de validation	DJF	SUITP	<5>
6.3.5.23	Documentation de test et de validation	DJF	SUITP	<5.3>
6.3.5.24	Documentation de test et de validation	DJF	SUITP	<5.5>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.3.5.25	Documentation de test et de validation	DJF	SUITP	<9.2>, <10>
6.2.7.9	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.11	Dossier des logiciels réutilisés	DJF	SRF	<9>
6.2.6.4	Fiches d'anomalies logicielles	DJF	-	
6.2.6.13.a	Rapport ISVV	DJF	-	
6.3.5.6	Fiches d'anomalies et fiches d'anomalies logicielles	DJF	-	
6.3.5.8	Fiches d'anomalies logicielles	DJF	-	
6.3.5.13	Rapports de test et de validation	DJF	-	
6.3.5.16	Documentation de test mise à jour	DJF	-	
6.3.5.17	Documentation de test mise à jour	DJF	-	
6.3.5.18	Documentation de test mise à jour	DJF	-	
6.3.5.28.b	Rapport ISVV	DJF	-	
6.3.5.30	Rapports de test et de validation	DJF	-	
6.3.5.31	Rapports de test et de validation	DJF	-	
7.3.6	Documentation de vérification et de validation des composants réutilisables	DJF	-	
7.3.7	Documentation de vérification et de validation des composants réutilisables	DJF	-	
6.2.4.4	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.5	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.8.b	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
7.2.2.3.b	Justification des choix de conception	DD	SDD	<4.5>

F.5 Résultats attendus de l'ECSS-Q-ST-80 - QR

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
5.2.1.3	Plan assurance produit logiciel	PAF	SPAP	Toutes
5.2.2.3	Bilans qualité logiciel de jalon	PAF	SPAMR	Toutes
6.2.3.3	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
6.2.6.12	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
5.5.5	Compte rendu des contrôles à la réception	PAF	-	

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.2.2.5	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.2.6	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.3.1.b	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.3.5.7	Déclaration de conformité aux plans et aux procédures de test	PAF	-	
6.3.5.11	Déclaration de conformité aux plans et aux procédures de test	PAF	-	
6.3.8.1	Plan de maintenance	MF	-	
6.3.8.2	Plan de maintenance	MF	-	
6.3.8.4	Plan de maintenance	MF	-	
6.3.8.5	Plan de maintenance	MF	-	
6.2.8.2	Documentation de validation et de test	DJF	SVS	<5.6>
6.2.8.7	Documentation de validation et de test	DJF	SVS	<5.6>
6.3.5.25	Documentation de test et de validation	DJF	SVS	<7.2>, <8>
6.3.5.29	Documentation de test et de validation	DJF	SVS	<6>
6.3.5.32	Spécification de validation du logiciel	DJF	SVS	<5>
6.2.6.5	Rapport de vérification logiciel	DJF	SVR	<4.4>
6.2.6.6	Rapport de vérification logiciel	DJF	SVR	<4.4>
7.1.7	Analyse de la précision numérique	DJF	SVR	<6>
7.2.3.6	Rapport de vérification logiciel	DJF	SVR	<4.5>
6.2.7.9	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.11	Dossier des logiciels réutilisés	DJF	SRF	<9>
6.2.6.4	Fiches d'anomalies logicielles	DJF	-	
6.2.6.13.a	Rapport ISVV	DJF	-	
6.3.5.6	Fiches d'anomalies et fiches d'anomalies logicielles	DJF	-	
6.3.5.8	Fiches d'anomalies logicielles	DJF	-	
6.3.5.13	Rapports de test et de validation	DJF	-	
6.3.5.16	Documentation de test mise à jour	DJF	-	
6.3.5.17	Documentation de test mise à jour	DJF	-	

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.3.5.18	Documentation de test mise à jour	DJF	-	
6.3.5.28.b	Rapport ISVV	DJF	-	
6.3.5.30	Rapports de test et de validation	DJF	-	
6.3.5.31	Rapports de test et de validation	DJF	-	
6.2.4.4	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.5	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.8.b	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes

F.6 Résultats attendus de l'ECSS-Q-ST-80 - AR

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
5.2.1.3	Plan assurance produit logiciel	PAF	SPAP	Toutes
5.2.1.4	Plan assurance produit logiciel	PAF	SPAP	<5.10>
5.2.2.3	Bilans qualité logiciel de jalon	PAF	SPAMR	Toutes
6.2.3.3	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
6.2.6.12	Bilans qualité logiciel de jalon	PAF	SPAMR	<4>
6.2.2.5	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.2.6	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.2.3.1.b	Rapport d'analyse de sûreté de fonctionnement et de sécurité des logiciels	PAF	-	
6.3.5.7	Déclaration de conformité aux plans et aux procédures de test	PAF	-	
6.3.5.11	Déclaration de conformité aux plans et aux procédures de test	PAF	-	
6.3.8.1	Plan de maintenance	MF	-	
6.3.8.2	Plan de maintenance	MF	-	
6.3.8.4	Plan de maintenance	MF	-	
6.3.8.5	Plan de maintenance	MF	-	
6.2.8.2	Documentation de validation et de test	DJF	SVS	<5.6>
6.2.8.7	Documentation de validation et de test	DJF	SVS	<5.6>

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
6.3.5.25	Documentation de test et de validation	DJF	SVS	<7.2>, <8>
6.3.5.29	Documentation de test et de validation	DJF	SVS	<6>
6.3.5.32	Spécification de validation du logiciel	DJF	SVS	<5>
6.2.6.5	Rapport de vérification logiciel	DJF	SVR	<4.4>
6.2.6.6	Rapport de vérification logiciel	DJF	SVR	<4.4>
7.2.3.6	Rapport de vérification logiciel	DJF	SVR	<4.5>
6.2.7.9	Dossier des logiciels réutilisés	DJF	SRF	<8>
6.2.7.11	Dossier des logiciels réutilisés	DJF	SRF	<9>
6.2.6.4	Fiches d'anomalies logicielles	DJF	-	
6.2.6.13.a	Rapport ISVV	DJF	-	
6.3.5.6	Fiches d'anomalies et fiches d'anomalies logicielles	DJF	-	
6.3.5.8	Fiches d'anomalies logicielles	DJF	-	
6.3.5.13	Rapports de test et de validation	DJF	-	
6.3.5.16	Documentation de test mise à jour	DJF	-	
6.3.5.17	Documentation de test mise à jour	DJF	-	
6.3.5.18	Documentation de test mise à jour	DJF	-	
6.3.5.27	Documentation de test et de validation	DJF	-	
6.3.5.28.b	Rapport ISVV	DJF	-	
6.3.5.30	Rapports de test et de validation	DJF	-	
6.3.5.31	Rapports de test et de validation	DJF	-	
6.3.6.3	Plan de test de recette	DJF	-	
6.3.6.7	Fiches d'anomalies	DJF	-	
6.3.6.8	Rapport de test de recette	DJF	-	
6.3.6.9	Rapport de test de recette	DJF	-	
6.2.4.4	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.5	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.8.b	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.3.6.1	Procédure d'installation	DD	SCF	<4.2>

F.7 Résultats attendus de l'ECSS-Q-ST-80 non associés à des revues de jalon spécifiques

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
5.1.5.2	Justificatifs relatifs à la formation et à l'expérience	PAF	-	
5.2.2.1	Bilan qualité logiciel	PAF	-	
5.2.2.2	Bilan qualité logiciel	PAF	-	
5.2.4.a	Avis d'alerte préliminaire	PAF	-	
5.2.4.b	Avis d'alerte	PAF	-	
5.4.1.1.a	Résultats des audits et des évaluations avant contrat	PAF	-	
5.4.1.1.b	Enregistrements des sources d'approvisionnement	PAF	-	
5.6.1.3	Bilans qualité logiciel	PAF	-	
5.7.1	Enregistrements des évaluations des processus logiciel : plan du programme général d'évaluation et d'amélioration	PAF	-	
5.7.2.1.a	Enregistrement des évaluations des processus logiciel : modèle d'évaluation	PAF	-	
5.7.2.1.b	Enregistrement des évaluations des processus logiciel : méthode d'évaluation	PAF	-	
5.7.2.2.a	Enregistrement des évaluations des processus logiciel : preuve de conformité au modèle d'évaluation des processus	PAF	-	
5.7.2.2.b	Enregistrement des évaluations des processus logiciel : méthode d'évaluation	PAF	-	
5.7.2.3	Enregistrement des évaluations des processus logiciel : preuve de reconnaissance des évaluations des processus logiciel	PAF	-	
5.7.2.4	Enregistrement des évaluations des processus logiciel : justification de l'expert compétent	PAF	-	
5.7.3.1	Enregistrements des évaluations des processus logiciel : plan d'amélioration	PAF	-	
5.7.3.2	Enregistrements des évaluations des processus logiciel : processus d'amélioration	PAF	-	

Paragraphe	Résultat attendu	Dossier cible	DRD	Section
5.7.3.3	Enregistrements des évaluations des processus logiciel : preuve des améliorations	PAF	-	
6.2.5.4	Bilans qualité logiciel	PAF	-	
6.2.5.5	Bilans qualité logiciel	PAF	-	
6.2.6.2	Bilans qualité logiciel	PAF	-	
6.2.6.3	Bilans qualité logiciel	PAF	-	
6.2.6.7	Bilans qualité logiciel	PAF	-	
6.2.6.9	Plans ou procédures de revue et de contrôle	PAF	-	
6.2.6.10	Plans ou procédures de revue et de contrôle	PAF	-	
6.2.6.11	Enregistrements relatifs aux revues et aux contrôles	PAF	-	
6.2.8.5	Bilans qualité logiciel	PAF	-	
6.3.3.4	Bilans qualité logiciel	PAF	-	
6.3.3.6	Bilans qualité logiciel	PAF	-	
6.3.3.7.b	Bilans qualité logiciel	PAF	-	
6.3.4.7	Bilans qualité logiciel	PAF	-	
6.3.5.3	Bilans qualité logiciel	PAF	-	
6.3.5.5	Bilans qualité logiciel	PAF	-	
6.3.5.12	Bilans qualité logiciel	PAF	-	
7.1.6	Bilans qualité logiciel	PAF	-	
7.1.7	Bilans qualité logiciel	PAF	-	
6.3.8.6	Enregistrements de maintenance	MF	-	
6.3.8.7	Enregistrements de maintenance	MF	-	
5.2.6.1.a.b	Fiches d'anomalies	DJF	-	
5.4.1.2	Dossier des logiciels réutilisés	DJF	SRF	Toutes
6.2.4.3.a	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.3.b	Document de version du logiciel	DD	SReID	Toutes
6.2.4.10	Dossier descriptif de configuration du logiciel	DD	SCF	Toutes
6.2.4.11.b	Marquages	DD	-	

Bibliographie

ECSS-S-ST-00	Système ECSS – Exigences générales, de description et de mise en œuvre
ECSS-Q-HB-80-02	Assurance produit des projets spatiaux – Evaluation et amélioration des processus logiciel
ECSS-Q-HB-80-03	Assurance produit des projets spatiaux – Méthodes et techniques d’analyse de sûreté de fonctionnement et de sécurité des logiciels
ECSS-Q-HB-80-04	Assurance produit des projets spatiaux – Définition et mise en œuvre du programme de métriques logiciel
ECSS-Q-ST-30-02	Assurance produit des projets spatiaux – Analyse des modes de défaillances, de leurs effets (et de leur criticité)
IEEE 610.12:1990	Glossaire des normes IEEE de la terminologie de l’ingénierie des logiciels (IEEE Standard Glossary of software engineering terminology)
IEEE 1028-1997	Norme IEEE relative aux revues logiciel (IEEE Standard for Software Reviews)
ISO 9000:2000	Systèmes de management de la qualité – Principes essentiels et vocabulaire
ISO 9126-1:2001	Génie du logiciel – Qualité des produits – Partie 1 : Modèle de qualité
ISO/IEC 12207:1995	Technologies de l’information – Processus du cycle de vie du logiciel
ISO/IEC 15504:1998	Technologies de l’information – Evaluation des procédés
RTCA/DO-178B	Considérations logicielles en matière de certification d’équipements et de systèmes embarqués (Software considerations in airborne systems and equipment certification)
CMU/SEI-92-TR-022	Mesure de la qualité logiciel : structure de comptabilisation des problèmes et des défauts (Software Quality Measurement: A framework for counting problems and defects)
CMU/SEI-2006-TR-008	CMMI pour développement, version 1.2