

Normes

Clémence Agrapart
clemence.agrapart@cns-orleans.fr

Lundi 16 octobre

Définitions

- **Norme** : « Ensemble de règles de conformité ou de fonctionnement légiféré par un organisme de normalisation mandaté : AFNOR, ISO, CEN, IEEE, ... »
- **Standard** : « Référentiel à diffusion large »
- **Référentiel** : « Ensemble de recommandations ou de préférences préconisées par un groupe d'utilisateurs caractéristiques et avisés »
- **ISO** (Normes fondamentales, normes de spécifications, normes d'analyse et d'essais, normes d'organisation): Organisation internationale de normalisation, non gouvernementale, indépendante, dont les 166 membres sont les organismes nationaux de normalisation.
- **IEC** : Commission electrotechnique internationale (« International Electrotechnical Commission »), organisation internationale de normalisation chargée des domaines de l'électricité, de l'électronique, de la compatibilité électromagnétique, de la nanotechnologie et des technologies connexes

Définitions

- **ECSS** : (European Cooperation for Space Standardization) sont des référentiels dédiés au management des projets spatiaux. Ils sont décomposés en quatre branches, elles-mêmes décomposées en disciplines
- **RNC** : (Référentiel Normatif du CNES) Le CNES a développé son propre système normatif. Ce système est constitué de documents issus principalement des ECSS. Il a été mis en place dans le but d'établir les spécifications des projets conduits par le CNES au travers de matrices d'applicabilité associées. Les documents RNC s'appliquent à toutes les phases et activités et sur toute la durée du projet, de l'analyse de la mission au retrait du service.

Définitions



Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12207
3. Modèle de qualité des produits logiciels : ISO15939
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 33001
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Tests : IEEE 829, 10044, 1008
8. Audit interne : CobIT

Quelles applications ?

Qualité système, relation client-fournisseur :

ISO 9000, ISO 9001, ISO 9004

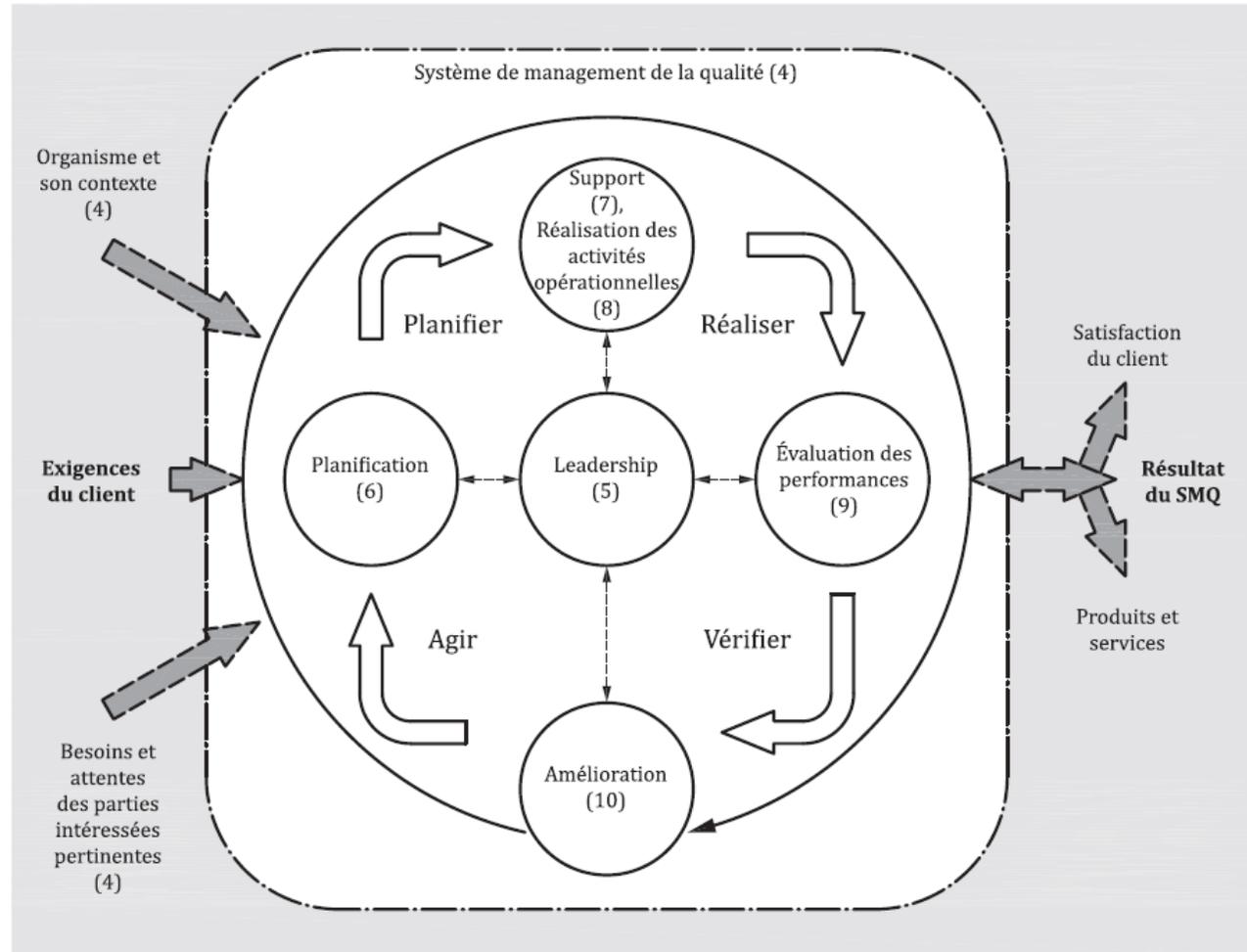
ISO 9000 : "Systèmes de management de la qualité - Principes essentiels et vocabulaire »

ISO 9001 : "Systèmes de management de la qualité – Exigences »

ISO 9004 fournit des lignes directrices permettant d'améliorer l'aptitude d'un organisme à obtenir des performances durables.

Quelles applications ?

ISO 9001_V2015 (extrait) :



NOTE Les nombres entre parenthèses font référence aux articles de la présente Norme internationale.

Quelles applications ?

Chapitre 4 : Contexte de l'organisme

→ Compréhension des besoins et des attentes des parties intéressées

Chapitre 5 : Leadership

→ Orientation client (exigences clients comprises)

Chapitre 6 : Planification

→ Actions à mettre en œuvre face aux risques et opportunités, planification des modifications

Chapitre 7 : Support

→ Fournir et maintenir l'infrastructure nécessaire à l'obtention de la conformité des produits et des services

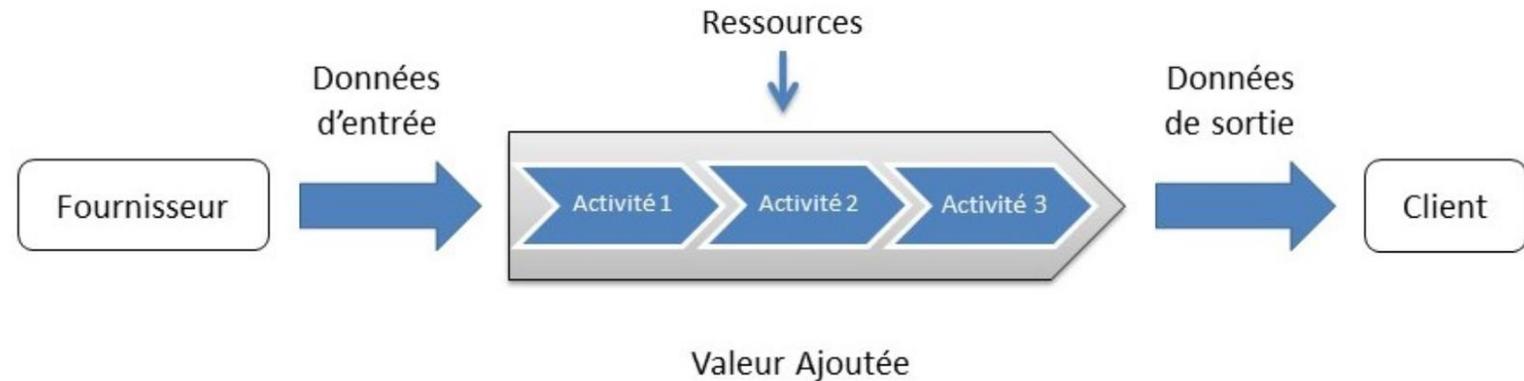
→ Compétences

→ Maîtrise des informations documentées

Quelles applications ?

Chapitre 8 : Réalisation des activités opérationnelles

→ Maîtrise des processus :



dont la maîtrise des processus externalisés

- Planification de la conception et du développement (éléments d'entrée, éléments de sortie, maîtrise des activités)
- Production et prestation des services
- Maîtrise des éléments non conformes

Quelles applications ?

Chapitre 9 : Evaluation des performances

- Déterminer ce qu'il est nécessaire de surveiller et mesurer
- Mesurer la satisfaction client (« surveiller la perception des clients sur le niveau de satisfaction de leurs besoins et attentes. »)
- Audit interne

Chapitre 10 : Amélioration

- Opportunités d'amélioration

Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, IEC25010, IEC 25000, ISO12207
3. Modèle de qualité des produits logiciels : ISO15939
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 33001
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Audit interne : CobIT

Quelles applications ?

- Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730
 - Organisation
 - Niveau de criticité du logiciel
 - Outils, techniques et méthodologies utilisées dans le processus d'AQL
 - Ressources employées, les normes, pratiques et conventions utilisées et les calendriers pertinents.
 - Détail des activités et tâches du cycle de vie de l'AQL en décrivant le rôle de l'assurance du produit et du processus
 - Description des activités et tâches
 - Revue de contrat et de mesure de la qualité
 - Politiques de test
 - Politique de dérogation et de déviation
 - Politique d'itération des tâches
 - Traçabilité
 - Un plan d'AQL doit décrire les activités qualité qui seront faites pour un projet ou une modification à un logiciel développé, documenté et maintenu à jour tout au long du projet

Quelles applications ?

- ISO/IEC 25000 :

Ingénierie des systèmes et du logiciel

Exigences de qualité des systèmes et du logiciel et évaluation (SQuaRE) — Guide de SQuaRE

- ISO/IEC 25010 :

Ingénierie des systèmes et logiciels

Exigences de qualité et évaluation des systèmes et du logiciel (SQuaRE) — Modèles de qualité du système et du logiciel

- ISO/IEC/IEEE 12207 :

Systems and software engineering — Software life cycle processes

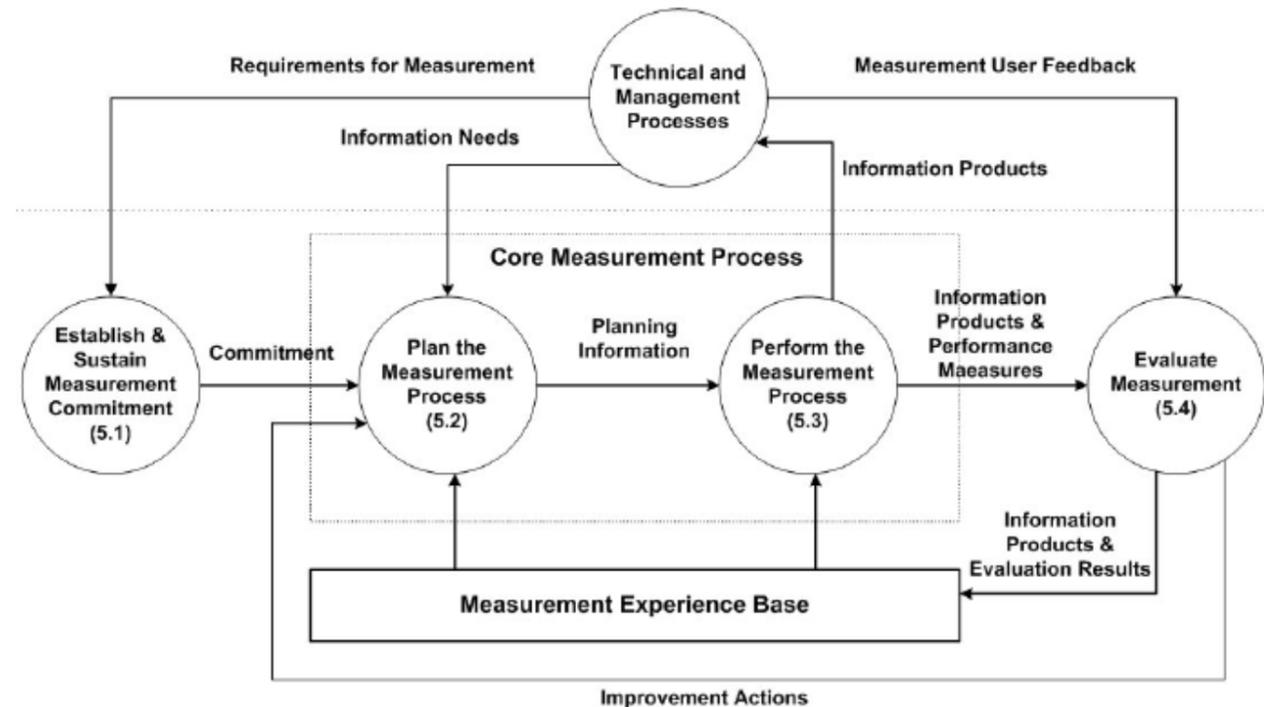
Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12207
3. **Modèle de qualité des produits logiciels : ISO15939**
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 33001
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Tests : IEEE 829, 10044, 1008
8. Audit interne : CobIT

Quelles applications ?

Modèle de qualité des produits logiciels :

- ISO/IEC/IEEE 15939:2017
- Ingénierie des systèmes et du logiciel — Processus de mesure



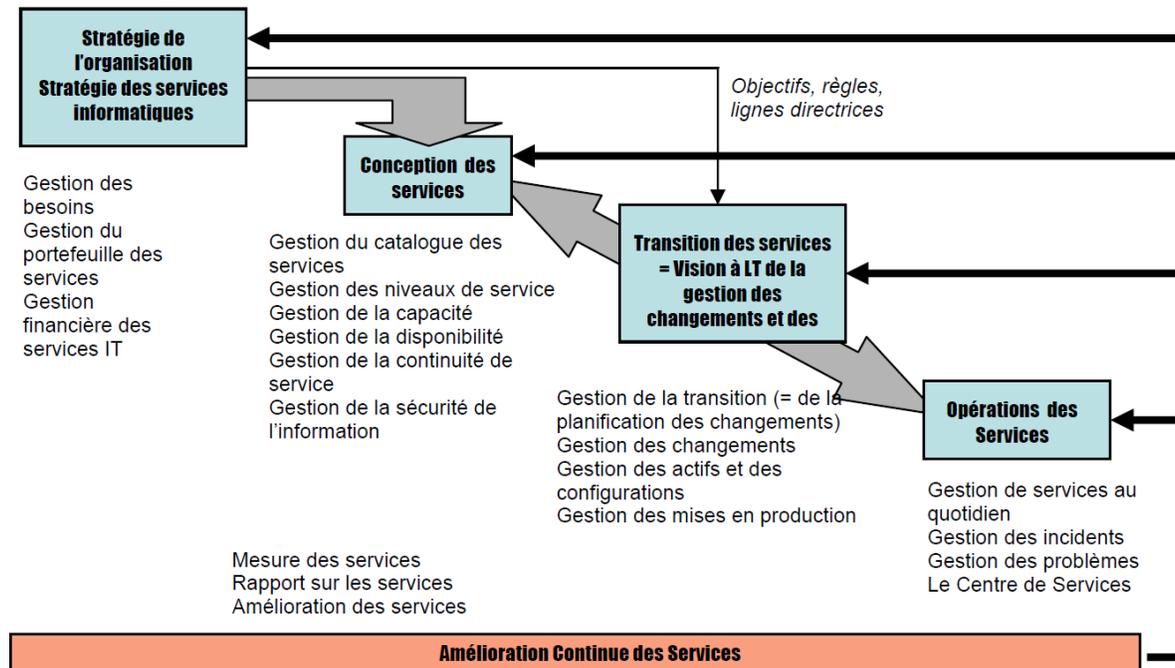
Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12207
3. Modèle de qualité des produits logiciels : ISO15939
4. **Gestion de service / infrastructure : ITIL et ISO20000**
5. Développement au sein d'une organisation : CMMi, ISO/IEC 33001
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Tests : IEEE 829, 10044, 1008
8. Audit interne : CobIT

Quelles applications ?

ITIL :

- Bonnes pratiques du management du système d'information.
- Référentiel qui aborde notamment :
 - Comment organiser un système d'information ?
 - Comment améliorer l'efficacité du système d'information ?
 - Comment réduire les risques ?
 - Comment augmenter la qualité des services informatiques ?



Quelles applications ?

ISO20000

- Issue de BSI (British Standards Institution)
- Norme de certification des services informatiques
- Comporte deux volets :
 - ISO/IEC 20000-1:2018 : Technologies de l'information -- Gestion des services -- Partie 1: Exigences du système de management des services ;
 - ISO/IEC 20000-2:2012 : Technologies de l'information -- Gestion des services -- Partie 2: Guide pour l'application des systèmes de management des services.

Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12207
3. Modèle de qualité des produits logiciels : ISO15939
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 33001
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Tests : IEEE 829, 10044, 1008
8. Audit interne : CobIT

Quelles applications ?

CMMi

- Le CMMI (Capability Maturity Model Integration), est un modèle qui permet de mesurer la maturité d'une organisation sur une échelle de 1 à 5.
- Initial → Géré → Défini → Géré quantitativement → En optimisation
- Principes basés sur de l'amélioration de performance (synthèse issue de l'industrie)
- Contient les pratiques clés et les cibles de chaque « niveau »

Quelles applications ?

ISO/IEC 33001

- **Software Process Improvement and Capability Determination (SPICE).**
- Capabilité / Processus

Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12207
3. Modèle de qualité des produits logiciels : ISO15939
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 33001
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Tests : IEEE 829, 10044, 1008
8. Audit interne : CobIT

Quelles applications ?

MISRA (Motor Industry Software Reliability Association)

- Elle spécifie les règles de programmation pour plus de sûreté. L'objectif est d'éviter des erreurs d'exécution provoquées par une construction de C incertaine, des faiblesses de structure du fait de malentendus entre les programmeurs, et de garantir la validité des expressions. La plupart de ces règles peuvent être automatiquement vérifiées ou prises en considération lors de la génération du code.

- **ISO/IEC 9899:2018**

Technologies de l'information — Langages de programmation — C

Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12207
3. Modèle de qualité des produits logiciels : ISO15939
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 15504
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. **Tests : IEEE 829, 1044, 1008**
8. Audit interne : CobIT

Quelles applications ?

- IEEE 829
 - *l'IEEE 829-2008* est le nom technique du standard de l'IEEE pour la documentation de test logiciel. Ce standard donne des spécifications pour la forme d'un ensemble de documents pour l'usage du test logiciel.
 - *Planning de test*. Ce document de planification de management montre : comment sera effectuée la campagne de test, qui la fera, ce qui sera testé, combien de temps cela prendra, quelle sera la couverture de test.
 - *Spécifications techniques de test*. Ce document détaille les conditions de test et les résultats attendus.
 - *Spécifications de cas tests*. Ce document spécifie les données de test pour l'exécution des conditions de tests définies dans le document précédent.
 - *Spécifications de procédure de test*. Ce document détaille comment chaque test est exécuté, notamment les pré-conditions de configuration et les scénarios de test.
 - *Rapport de transmission de cas tests*. Ce document rapporte le passage d'une étape à une autre d'un scénario de test.
 - *Journal de test*. Ce document enregistre les cas tests exécutés, leur auteur, leur ordre d'exécution, et le résultat du cas test.
 - *Rapport d'incidents de test*. Ce document détaille pour chaque test non validé, les résultats attendu et obtenu, ainsi que des informations pour comprendre la raison du décalage. Ce document s'appelle bien un rapport d'incidents et non de fautes. En effet, les raisons du décalage entre le résultat attendu et le résultat obtenu ne sont pas toujours le fait d'une faute du système. Il est par contre exclu que le résultat de test soit obtenu du fait d'une mauvaise exécution du test ou d'exigences testables équivoques. Ce rapport entre dans le détail des incidents de tests : quand l'incident survient, quelles remarques peuvent aider à sa résolution. Si possible, le rapport inclut aussi une évaluation de l'impact de l'incident sur la suite de la campagne de test.
 - *Rapport de test (ou PV de recette)*. Ce document de management fournit les informations importantes mises au jour par la campagne de tests exécutée : une évaluation de la qualité de l'effort de test, de la qualité du système testé, des statistiques du rapport d'incidents de test. Le rapport comporte aussi la durée des différents tests menés, et comment améliorer une prochaine campagne de test. Ce document final indique si le système testé est approuvé ou pas, et selon quels critères.

Quelles applications ?

- IEEE 1044
 - définit comment classer les anomalies. Ces anomalies des **tests de logiciels** sont communément appelés des erreurs ou des bugs.
- IEEE 1008
 - est la **norme** pour les **tests** unitaires du **logiciel**.
- ISO 25010 :
 - Tests fonctionnels : exactitude, complétude, aptitude à l'usage
 - de performance : temps de réponse, utilisation de ressources, capacité
 - de compatibilité : coexistence, interopérabilité
 - d'utilisabilité : intelligibilité, apprentissage, ergonomie, accessibilité
 - fiabilité : maturité, disponibilité, robustesse, récupérable
 - Sécurité : confidentialité, intégrité, responsabilité, authenticité
 - Maintenabilité : modularité, réutilisabilité, modifiable, testable
 - Portabilité : adaptabilité, installable, remplaçable

Quelles applications ?

1. Qualité système, relation client-fournisseur : ISO 9000, ISO 9001, ISO 9004
2. Plan AQL (Assurance Qualité Logiciel) : Norme IEEE 730, ISO25010, ISO 25000, ISO12007
3. Modèle de qualité des produits logiciels : ISO15939
4. Gestion de service / infrastructure : ITIL et ISO20000
5. Développement au sein d'une organisation : CMMi, ISO/IEC 15504
6. Règles de codage (lecture, modification, portabilité) : MISRA
7. Tests : IEEE 829, 1044, 1008
8. **Audit interne : CobIT**

Quelles applications ?

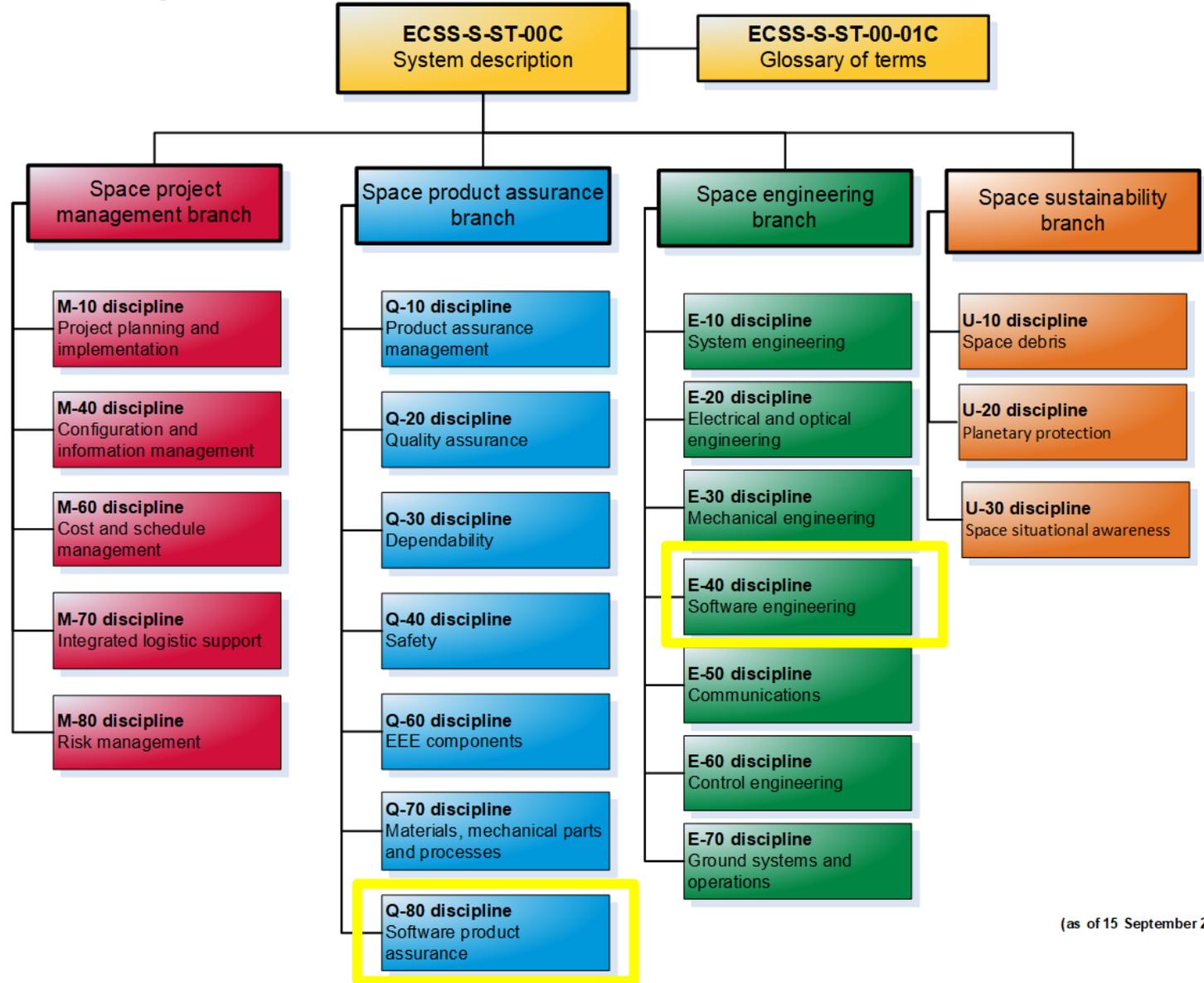
Audit interne : CobIT

Critères liés à l'information/donnée :

- Efficacité
- Efficience
- Confidentialité
- Intégrité
- Disponibilité
- Conformité
- Fiabilité

Point sur les ECSS

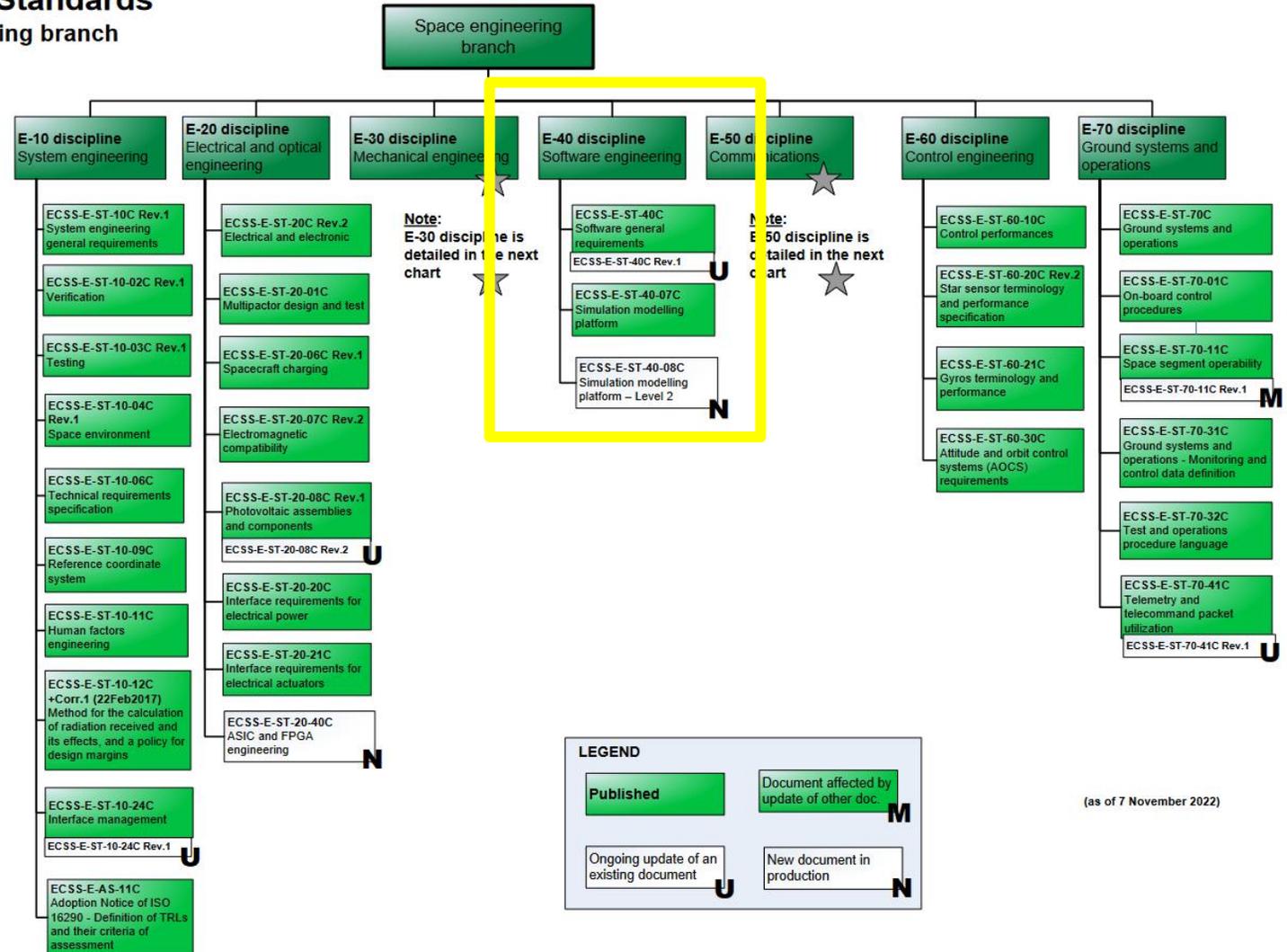
ECSS Disciplines



(as of 15 September 2021)

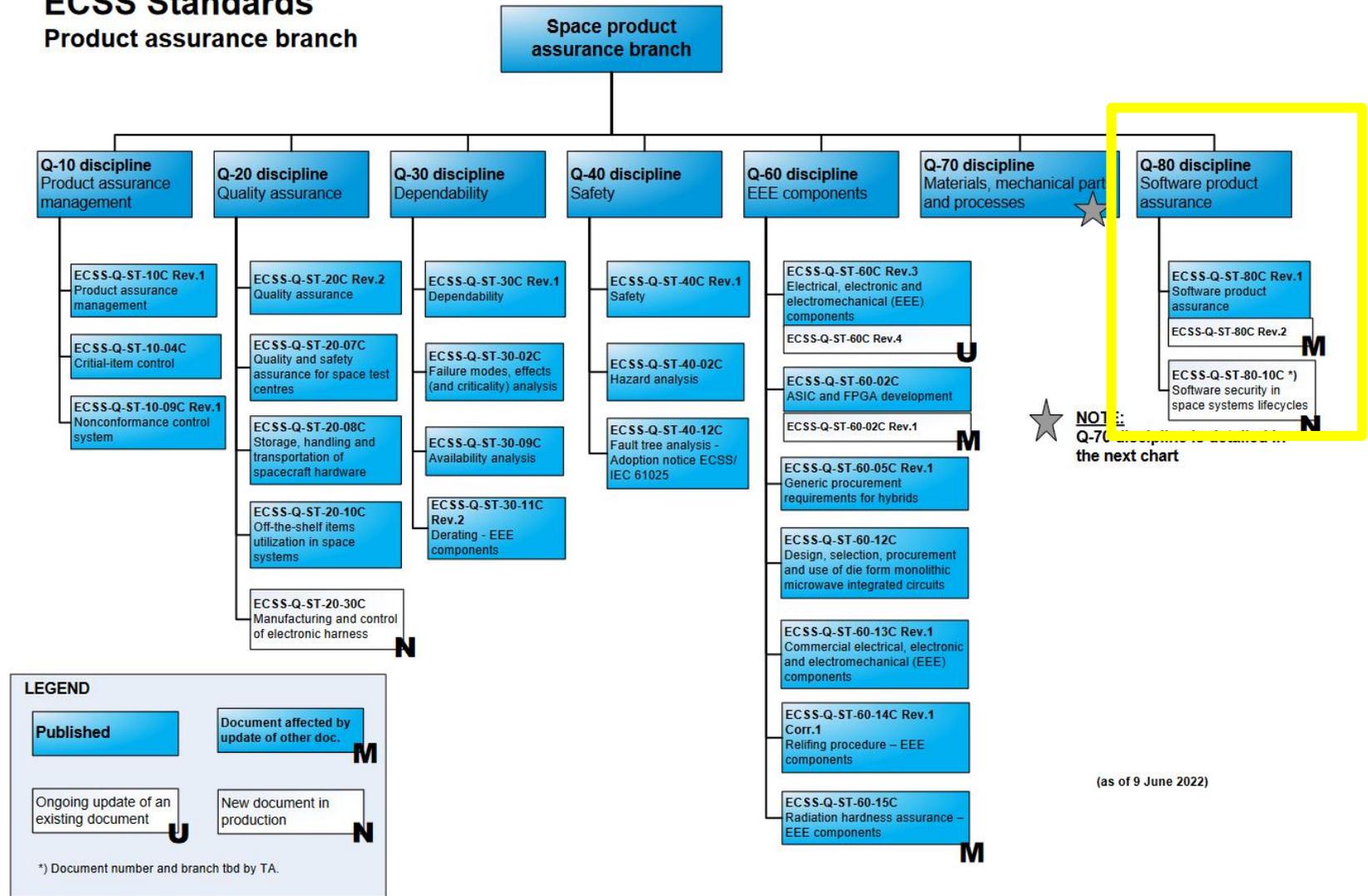
Point sur les ECSS

ECSS Standards Engineering branch



Point sur les ECSS

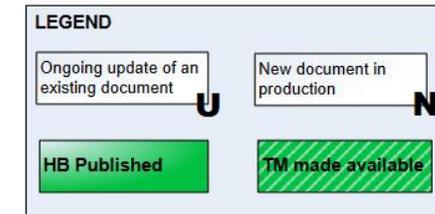
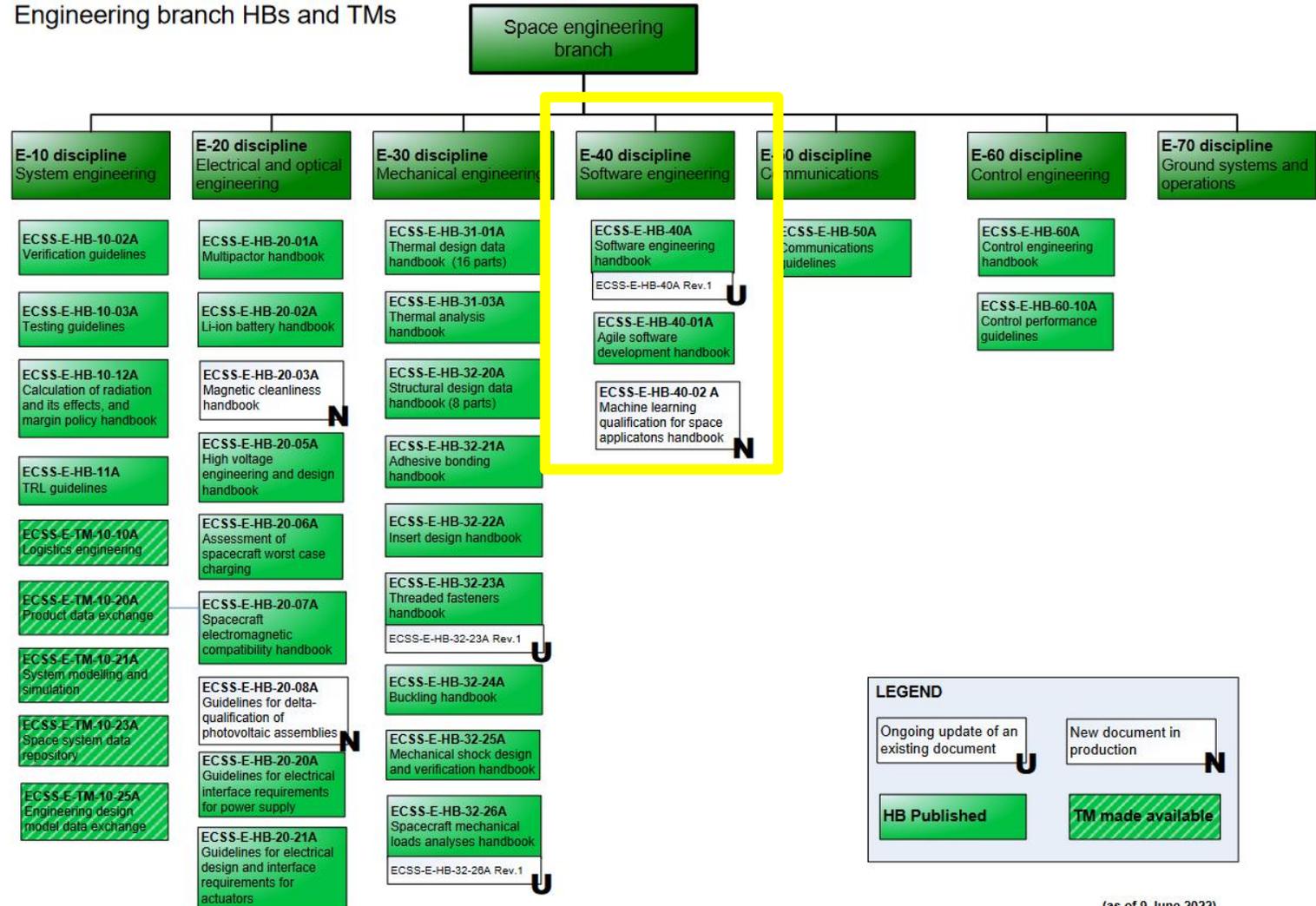
ECSS Standards Product assurance branch



Point sur les ECSS

ECSS Handbooks and Technical memoranda

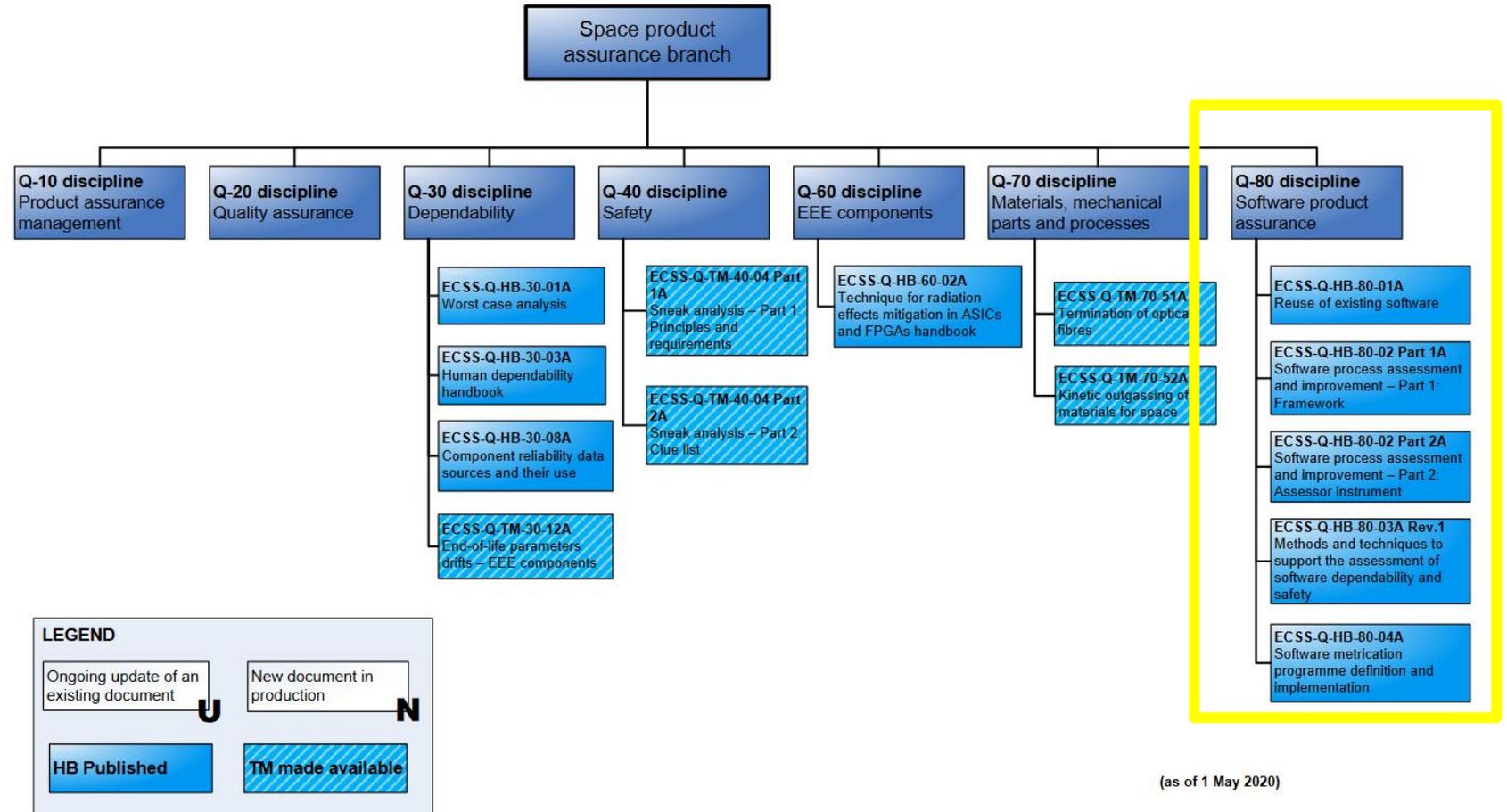
Engineering branch HBs and TMs



(as of 9 June 2022)

Point sur les ECSS

ECSS Handbooks and Technical memoranda PA branch HBs and TMs



Exemples de documents RNC (CNES)

<u>INGENIERIE LOGICIEL</u>	<u>REGLES D'UTILISATION DES LANGAGES</u>	<u>ASSURANCE QUALITE LOGICIEL</u>
<ul style="list-style-type: none"> • Logiciels : Principes et exigences [RNC-ECSS-E-ST-40] • Règles et recommandations pour conduire une recette de produit logiciel [RNC-CNES-E-HB-40-506] • Contenu type de la Spécification Technique de Besoin Logiciel (STBL) [RNC-CNES-E-HB-40-501] • Règles pour l'élaboration de documents de description d'interfaces informatiques [RNC-CNES-E-HB-40-507] • Manuel utilisateur et manuel d'exploitation d'un système informatique sol de traitement [RNC-CNES-E-HB-40-503] • Démarche de développement objet pour les logiciels [RNC-CNES-E-HB-40-508] <ul style="list-style-type: none"> + Principes de mise en œuvre des concepts objets [508-A] + Documentation [508-B] + Terminologie orientée objet [508-C] + Exemple de développement orienté objet [508-D] • Règles et recommandations d'ergonomie des IHM informatiques [RNC-CNES-E-HB-40-504] • Règles et recommandations pour l'utilisation du formalisme UML [RNC-CNES-E-HB-40-509] • Règles et recommandations pour la réalisation d'un serveur web [RNC-CNES-E-HB-40-505] <ul style="list-style-type: none"> + Règles pour la réalisation de chartes pour un serveur web [505-A] 	<ul style="list-style-type: none"> • Règles communes pour l'utilisation des langages de programmation [RNC-CNES-Q-ST-80-501] • Règles pour l'utilisation du langage C [RNC-CNES-Q-HB-80-506] <ul style="list-style-type: none"> + Règles complémentaires pour l'utilisation du langage C dans les logiciels embarqués [506-A] • Règles pour l'utilisation du langage C++ [RNC-CNES-Q-HB-80-513] • Règles pour l'utilisation du langage Java [RNC-CNES-Q-80-527] • Règles pour l'utilisation du langage IDL [RNC-CNES-Q-HB-80-534] • Règles pour l'utilisation du langage Python [RNC-CNES-Q-HB-80-535] 	<ul style="list-style-type: none"> • Assurance produit Logiciel : exigences pour le développement et la maintenance des logiciels [RNC-ECSS-Q-ST-80] • Exigences AQ simplifiées pour le développement logiciel [RNC-CNES-Q-ST-80-100] • Guide : Sélection et interprétation des mesures de complexité Logiciel [RNC-CNES-Q-ST-80-503] • Contenu d'un bilan qualité logiciel [RNC-CNES-Q-ST-80-510] • Contenu d'un bilan de projet logiciel [RNC-CNES-Q-ST-80-511] • Contenu d'un plan de contrôle de la qualité d'un logiciel [RNC-CNES-Q-ST-80-515]

Exemples de documents RNC (CNES)

- Logiciels :
Principes et exigences (RNC-ECSS-E-ST-40)

Historique des modifications	4
Introduction.....	12
1 Domaine d'application	13
2 Références normatives	14
3 Termes, définitions et abréviations	15
3.1 Termes définis dans d'autres normes	15
3.2 Termes spécifiques à la présente norme	15
3.3 Abréviations.....	21
4 Principes d'assurance produit des logiciels des systèmes spatiaux.....	24
4.1 Introduction.....	24
4.2 Présentation générale des processus d'ingénierie des logiciels des systèmes spatiaux	25
4.2.1 Généralités	25
4.2.2 Traitement des exigences système relatives au logiciel	28
4.2.3 Processus de management du logiciel.....	29
4.2.4 Processus d'ingénierie de l'architecture et des exigences logiciel.....	29
4.2.5 Processus d'ingénierie de conception et de production du logiciel	29
4.2.6 Processus de validation du logiciel.....	30
4.2.7 Processus de livraison et d'acceptation du logiciel.....	30
4.2.8 Processus de vérification du logiciel.....	31
4.2.9 Processus d'exploitation du logiciel.....	31
4.2.10 Processus de maintenance du logiciel	32
4.3 Structure de la présente norme.....	33
4.4 Adaptation de la présente norme.....	35
5 Exigences.....	36
5.1 Introduction.....	36
5.2 Traitement des exigences système relatives au logiciel.....	37

Exemples de documents RNC (CNES)

- Logiciels :
Principes et exigences (RNC-ECSS-E-ST-40)

5.2.1	Présentation générale.....	37
5.2.2	Analyse des exigences système relatives au logiciel.....	37
5.2.3	Vérification système relative au logiciel.....	38
5.2.4	Intégration et contrôle système relatifs au logiciel.....	39
5.2.5	Revue des exigences de besoin.....	41
5.3	Processus de management du logiciel.....	41
5.3.1	Présentation générale.....	41
5.3.2	Gestion du cycle de vie du logiciel.....	41
5.3.3	Processus de revue commune.....	43
5.3.4	Description des revues de projet logiciel.....	45
5.3.5	Description des revues techniques de logiciel.....	46
5.3.6	Logique de déroulement des revues.....	47
5.3.7	Gestion des interfaces.....	48
5.3.8	Gestion des marges et des budgets techniques.....	48
5.3.9	Conformité à la présente norme.....	49
5.4	Processus d'ingénierie de l'architecture et des exigences logiciel.....	50
5.4.1	Présentation générale.....	50
5.4.2	Analyse des exigences logiciel.....	50
5.4.3	Conception d'architecture du logiciel.....	52
5.4.4	Réalisation d'une revue de conception préliminaire.....	54
5.5	Processus d'ingénierie de conception et de production du logiciel.....	54
5.5.1	Présentation générale.....	54
5.5.2	Conception des éléments logiciel.....	54
5.5.3	Codage et essai.....	57
5.5.4	Intégration.....	58
5.6	Processus de validation du logiciel.....	59
5.6.1	Présentation générale.....	59
5.6.2	Mise en œuvre du processus de validation.....	59
5.6.3	Activités de validation par rapport à la spécification technique.....	60
5.6.4	Activités de validation par rapport aux exigences de besoin.....	61
5.7	Processus de livraison et d'acceptation du logiciel.....	63
5.7.1	Présentation générale.....	63
5.7.2	Livraison et installation du logiciel.....	63
5.7.3	Acceptation du logiciel.....	64
5.8	Processus de vérification du logiciel.....	65
5.8.1	Présentation générale.....	65

Exemples de documents RNC (CNES)

- Logiciels :
Principes et exigences (RNC-ECSS-E-ST-40)

Annexe O (normative) Plan de développement logiciel (SDP) – DRD	177
Annexe P (normative) Plan de revue du logiciel (SRevP) – DRD	184
Annexe Q (informative) Organisation et contenu des documents à chaque jalon	194
Annexe R (normative) Adaptation de la présente norme en fonction de la criticité du logiciel	213

5.8.2	Mise en œuvre du processus de vérification	65
5.8.3	Activités de vérification	66
5.9	Processus d'exploitation du logiciel	75
5.9.1	Présentation générale	75
5.9.2	Mise en œuvre du processus	75
5.9.3	Essais fonctionnels	76
5.9.4	Support d'exploitation du logiciel	77
5.9.5	Soutien aux utilisateurs	77
5.10	Processus de maintenance du logiciel	78
5.10.1	Présentation générale	78
5.10.2	Mise en œuvre du processus	78
5.10.3	Analyse des problèmes et des modifications	79
5.10.4	Application des modifications	80
5.10.5	Réalisation de revues de maintenance	81
5.10.6	Migration du logiciel	81
5.10.7	Retrait du logiciel	83
Annexe A (informative) Documentation du logiciel		85
Annexe B (normative) Spécification de besoin du logiciel (SSS) – DRD		92
Annexe C (normative) Spécification de gestion des interfaces logiciel (IRD) – DRD		102
Annexe D (normative) Spécification des exigences logiciel (SRS) – DRD		106
Annexe E (normative) Document de contrôle des interfaces (ICD) – DRD		114
Annexe F (normative) Document de conception du logiciel (SDD) – DRD		118
Annexe G (normative) Document de version du logiciel (SReID) – DRD		129
Annexe H (normative) Manuel utilisateur du logiciel (SUM) – DRD		132
Annexe I (normative) Plan de vérification du logiciel (SVerP) – DRD		138
Annexe J (normative) Plan de validation du logiciel (SVaIP) – DRD		143
Annexe K (normative) Plan [de tests unitaires/d'essai d'intégration] du logiciel (SUITP) – DRD		148
Annexe L (normative) Spécification de validation du logiciel (SVS) – DRD		157
Annexe M (normative) Rapport de vérification du logiciel (SVR) – DRD		165
Annexe N (normative) Dossier des logiciels réutilisés (SRF) – DRD		173

Exemples de documents RNC (CNES)

- Assurance Produit Logiciel :

Exigences pour le développement et la maintenabilité des logiciels
(RNC-ECSS-Q-ST-8o)

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

Exemples de documents RNC (CNES)

- Assurance Produit Logiciel :

Exigences pour le développement et la maintenabilité des logiciels (RNC-ECSS-Q-ST-8o)

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

Exemples de documents RNC (CNES)

- Assurance Produit Logiciel :
Exigences pour le développement et la maintenabilité des logiciels (RNC-ECSS-Q-ST-8o)

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

Exemples de documents RNC (CNES)

Les RNC suivantes proposent des exigences « synthétiques » pour le développement, le suivi (ingénierie) et la maintenance de logiciels spatiaux :

- **RNC-CNES-Q-ST-80-100** : 24 exigences sur une base ECSS de 313 exigences
- **RNC-CNES-E-ST-40-110** : 42 exigences sur une base ECSS de 202 exigences

→ Elles offrent la possibilité d'un niveau d'exigences au moins équivalent à celui des ECSS.

→ Elles sont applicables aussi bien aux logiciels bord ou sol en développement ou maintenance. Elles sont adaptées à tout cycle de développement (V, itératif incrémental, agile, ...).

→ Elles permettent de faciliter les échanges et de gagner du temps dans les échanges client/fournisseur. La marche de prise en main est moins haute.

Où trouver les documents ?

AFNOR : <https://www.boutique.afnor.org/>

ISO/IEC : <https://www.iso.org/fr/news/ref2326.html>

ECSS : <https://ecss.nl/>

RNC : *sur demande auprès du CNES (interlocuteur projet par exemple)*

La composition du référentiel : RNC-GEST-01-LI-009

GNS : <https://gns.cnes.fr/fr/>

Et notamment : <https://gns.cnes.fr/fr/qualite-logiciel-ap-7>

Et pour aller
plus loin...

- D'autres documents utiles :
 - ISO/IEC 27001 : Cette norme traite de la gestion de la sécurité de l'information et peut être pertinente pour la qualité logicielle, en particulier pour les logiciels qui traitent des données sensibles. Elle met l'accent sur la gestion des risques de sécurité.
 - OWASP (Open Web Application Security Project) : Bien qu'il ne s'agisse pas d'une norme au sens formel, OWASP propose des bonnes pratiques et des outils pour améliorer la sécurité des applications web.
 - En cours :
Révision du guide d'application « GA X50-552 » et « NFX50- 553 »
Cette révision a pour objectif de donner des éléments pour permettre l'appropriation et la mise en œuvre d'un SMQ dans le domaine de la recherche. Il ne se substitue pas à la norme NF EN ISO 9001 mais explicite et illustre sa mise en œuvre en cohérence avec les termes et les pratiques de la recherche publique.
- <https://qualite-en-recherche.cnrs.fr/gt/qualite-en-recherche-inter-reseaux-qir-et-afnor-en-cours/>

Et pour aller plus loin...

- Vous pouvez aussi en interne définir des référentiels de « bonnes pratiques »
 - A travers la mise en place d'outils (JIRA, Outil GED, SVN, ...)



- A travers la mise en place de procédure ou référentiel laboratoire inspirés de procédure ou référentiel CNRS
 - Manuel conduite de projet (réseau MAPI : <https://mapi.cnrs.fr/documents-a-partager/groupe-de-travail-guide-de-bonnes-pratiques-de-la-conduite-de-projet/>)
 - Procédures INSU (réseau Qualité INSU)
 - DMP OPIDoR (plan de gestion de logiciel) : <https://dmp.opidor.fr/>
 - Cahier de laboratoire (*GT CdL du réseau QeR dématérialisé en cours*)
 - <https://www.devlog.cnrs.fr/>

Et pour aller
plus loin...

- **SMP (Software Management Plan) :**
Centralise les informations concernant le logiciel
Aide à la réflexion et la planification
- Au niveau équipe :
 - Une vue d'ensemble de chaque logiciel
 - Un suivi facile dans l'équipe
 - Amélioration des méthodes de gestion (gain de temps)
 - Identification d'une stratégie
- Au niveau laboratoire
 - Meilleure connaissance des productions scientifiques / Vue d'ensemble
 - Mise en œuvre d'une politique commune

Et pour aller plus loin...

- Exemple de SMP (Software Management Plan) au LPC2E :

Objectifs

- Se poser les questions en amont et ensemble sur les développements logiciels et la génération et gestion des données pour limiter les risques liés au turnover en réunissant les développeurs, l'encadrant, le nouvel entrant, les ASR.
- Promouvoir les pratiques et les outils labo et veiller à une certaine cohérence des développements.
- Aider à formaliser les besoins auprès des ASR/informatique appliquée/responsables ou identifier des besoins qui n'auraient pas été exprimés.

	Document de gestion de soft/données	LAB-REA-EER-FI-026-LPC2E V1.2 Page 1 / 8
--	-------------------------------------	--

Circuit nouvel entrant – rencontre avec les développeurs

Et pour aller plus loin...

- Exemple de SMP (Software Management Plan) au LPC2E :

	Document de gestion de soft/données	LAB-REA-EER-FI-026-LPC2E V1.2 Page 3 / 8
---	-------------------------------------	--

Ce document doit s'adapter au projet et à ses besoins (développement, traitement ou génération de données).

Il est composé d'un tableau avec les principales questions à se poser et les éléments de réponse attendus sur le sujet. Il peut être accompagné d'un logigramme représentant le processus résumé de gestion des traitements de données. La figure 1 présente un exemple de logigramme correspondant aux données atmosphère au laboratoire.

Pour toutes les rubriques, garder à l'esprit les besoins de collaboration avec des collaborateurs extérieurs et/ou les besoins de travail depuis l'extérieur (nomadisme, télétravail).

Rubrique	Item	Questions à se poser et éléments de réponse
Projet	Nom du projet	
	Description courte / Contexte	
	But du code	A quoi sert ce code ? (réponse succincte) <i>Code de modélisation, de traitement de données, de génération d'archives</i>
	Contenu des données	Quelles données sont générées ? (réponse succincte) <i>Données d'instrument, sorties de modèles, données internes, données officielles d'un projet devant servir d'entrée</i> <i>Données à livrer à un centre externe, à distribuer depuis le labo.</i>
	Utilisateur visé	A qui va servir ce code/ces données ? (réponse succincte) <i>Utilisation interne, code distribué/partagé avec la communauté, code livrable pour intégration dans un projet de plus grande ampleur.</i> <i>Code qui servira à d'autres projets, à d'autres équipes</i> Comment va être utilisé ce code/ces données ? (réponse succincte) <i>Code qui doit tourner sur une machine projet, sur un PC, un centre de calcul, une machine externe. Doit-il pouvoir être lancé par d'autres équipes ?</i>
	Lien avec un contrat (ou un brevet) ? ➔ Conséquences en termes de livrables,	Quels sont les livrables et le planning de livraison ? <i>Identifier les livrables (codes, données, documentation...) et les responsabilités</i>

Et pour aller
plus loin...

- Exemple de SMP (Software Management Plan) au LPC2E :

- Questions sur :
 - L'équipe
 - Les entrées et dépendances (OS+Licence / Langage+Compilateur+Licence..)
 - Données (d'entrée, générées,...)
 - Cible technique (versionnement code, stockage données, machine de développement, ...)
 - Vie du code/Données (livraison externe, diffusion vers l'extérieur, maintenance, support ...)
 - Documentation (mise à jour de ce document, documentation projet, ...)
 - Valorisation
 - ...

→ Ccl: La qualité logiciel dans son ensemble : au delà de la protection et de la reproductibilité de la donnée, pensez « qualité de la donnée » : robustesse et fiabilité

Bibliographie

Guide Assurance Produit CNRS (QeR_GT_APAQ) :

<https://qualite-en-recherche.cnrs.fr/guide/>

ISO/IEC : <https://www.iso.org/fr/news/ref2326.html>

Wikipedia :

https://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal

DMP OPIDoR (plan de gestion de logiciel) : <https://dmp.opidor.fr/>

Merci pour votre attention !

Clémence Agrapart
clemence.agrapart@cns-orleans.fr

Lundi 16 octobre