

Cours Gestion de projet

Normalisation des processus

Version	V2.3
Date	Janvier 2022
Auteur	Jean-Pierre ROBERTI

Normalisation des processus

Ce document est publié sous la licence libre **Creative Commons-BY-NC-SA**
<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>



BY : Paternité. Vous devez citer le nom de l'auteur original.

NC : Pas d'Utilisation Commerciale. Vous n'avez pas le droit d'utiliser cette création à des fins lucratives et commerciales.

SA : Partage des Conditions Initiales à l'Identique. Si vous modifiez, transformez ou adaptez cette création, vous n'avez le droit de distribuer la création qui en résulte que sous un contrat identique à celui-ci. En outre, à chaque réutilisation ou distribution, vous devez faire apparaître clairement aux autres les conditions contractuelles de mise à disposition de cette création.

Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits.

Normalisation des processus

- Sommaire
 - Bonne et mauvaise nouvelle
 - Les raisons et solutions
 - Processus logiciel
 - Assurance qualité logiciel
 - Référentiels qualité : Normes et modèles
 - ISO 9001:2015
 - CMMi
 - ISO CMMi / Synthèse
 - Exercices

La bonne nouvelle...



La mauvaise nouvelle...

“74% des projets échouent.”

Standish Group, CHAOS Report, 1999

La réussite des projets Temps/Budget/Résultats.....

Le chemin est encore long pour atteindre la pleine réussite...

MODERN RESOLUTION FOR ALL PROJECTS

	2011	2012	2013	2014	2015
SUCCESSFUL	29%	27%	31%	28%	29%
CHALLENGED	49%	56%	50%	55%	52%
FAILED	22%	17%	19%	17%	19%

The Modern Resolution (OnTime, OnBudget, with a satisfactory result) of all software projects from FY2011-2015 within the new CHAOS database. Please note that for the rest of this report CHAOS Resolution will refer to the Modern Resolution definition not the Traditional Resolution definition.

Meilleure probabilité de réussite pour les petits Projets

CHAOS RESOLUTION BY PROJECT SIZE

	SUCCESSFUL	CHALLENGED	FAILED
Grand	2%	7%	17%
Large	6%	17%	24%
Medium	9%	26%	31%
Moderate	21%	32%	17%
Small	62%	16%	11%
TOTAL	100%	100%	100%

The resolution of all software projects by size from FY2011-2015 within the new CHAOS database.

Méthodes AGILE plus performantes que CASCADE

CHAOS RESOLUTION BY AGILE VERSUS WATERFALL

SIZE	METHOD	SUCCESSFUL	CHALLENGED	FAILED
All Size Projects	Agile	39%	52%	9%
	Waterfall	11%	60%	29%
Large Size Projects	Agile	18%	59%	23%
	Waterfall	3%	55%	42%
Medium Size Projects	Agile	27%	62%	11%
	Waterfall	7%	68%	25%
Small Size Projects	Agile	58%	38%	4%
	Waterfall	44%	45%	11%

The resolution of all software projects from FY2011-2015 within the new CHAOS database, segmented by the agile process and waterfall method. The total number of software projects is over 10,000

Dépassements projets

Cost Overruns	% of Responses
Under 20%	15.5%
21 - 50%	31.5%
51 - 100%	29.6%
101 - 200%	10.2%
201 - 400%	8.8%
Over 400%	4.4%

Time Overruns	% of Responses
Under 20%	13.9%
21 - 50%	18.3%
51 - 100%	20.0%
101 - 200%	35.5%
201 - 400%	11.2%
Over 400%	1.1%

% of Features/Functions	% of Responses
Less Than 25%	4.6%
25 - 49%	27.2%
50 - 74%	21.8%
75 - 99%	39.1%
100%	7.3%

31,1% des projets sont **arrêtés**
52,7% des projets **dépassent** de 189% leurs prévisions budgétaires
16,2% des projets finissent dans les clous du point de vue du budget et des délais (#9% grandes entreprises)

Pourquoi les projets échouent-ils?

- Mauvaise compréhension des attentes et objectifs d'affaires
- Mauvaise compréhension de l'envergure, des fonctions et des livrables du système
- Faiblesse dans l'évaluation de l'envergure et des estimations
- Faiblesse dans la planification et le contrôle
- Structure décisionnelle inadéquate
- Manque de soutien de la Direction
- Communication inadéquate entre la direction de l'entreprise et le projet
- Mauvaise compréhension des risques
- Mauvaise évaluation de la capacité des ressources
- Faiblesse dans la gestion des changements et des problèmes
- Manque de culture partagé ou partage de culture manqué...
- ...

Pourquoi les projets réussissent-ils?

- Approche métier du chef de projet
- Définition claire des objectifs du projet
- Stratégie adéquate
- Planification, réalisation réaliste
- Contrôle adéquat
- Communication efficace
- Engagement de la direction et de l'équipe
- Partage d'une culture commune
- Mécanisme de décision efficace

FACILE ?

Facteurs de succès des projets

Project Success Factors	% of Responses
1. User Involvement	15.9%
2. Executive Management Support	13.9%
3. Clear Statement of Requirements	13.0%
4. Proper Planning	9.6%
5. Realistic Expectations	8.2%
6. Smaller Project Milestones	7.7%
7. Competent Staff	7.2%
8. Ownership	5.3%
9. Clear Vision & Objectives	2.9%
10. Hard-Working, Focused Staff	2.4%
Other	13.9%

La Solution ?

- Il faut **normaliser et maîtriser les processus** :

- Les définir selon l'état de l'art
- Les respecter
- Les améliorer

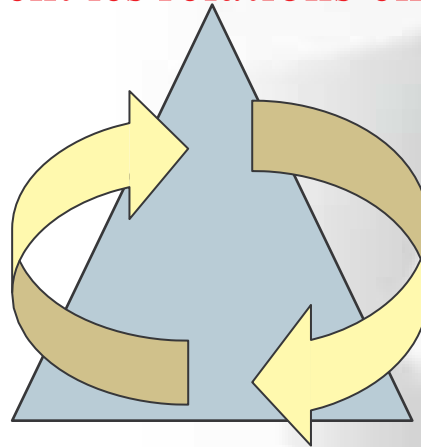


**«Dîtes ce que vous faites,
faites ce que vous dites,
et montrez que vous l'avez fait»**

Définition d'un **Processus Logiciel**

Un ensemble organisé d'activités, de méthodes, de pratiques et de transformations que des personnes utilisent pour développer et maintenir des logiciels et des produits associés à partir d'informations, d'énergie, d'équipements et d'outils.

Des procédures, des méthodes qui définissent les relations entre les tâches



Des personnes, leurs compétences, leur formation, leur motivation...

Des outils, des équipements ...

Définition d'un **Modèle Qualité**

- **Qu'est-ce qu'un modèle ?**

- Un modèle est un ensemble structuré d'éléments qui décrit les caractéristiques d'un processus éprouvé

- **Avantages :**

- Vocabulaire commun, vision partagée
 - Bâti sur des processus et pratiques de la communauté logicielle
 - Permet prioritisations et évaluations

- **Inconvénients :**

- Simplification du monde réel
 - Peu compréhensible parfois
 - Interprétation et adaptations basées sur objectifs business
 - Il faut y appliquer du jugement

Définition d'une Norme Qualité

- **Qu'est-ce qu'une Norme ?**

- Document établi par consensus par un organisme reconnu qui fournit des règles, des lignes directrices ou des caractéristiques pour les activités ou des résultats garantissant un niveau d'ordre optimal (qualité)

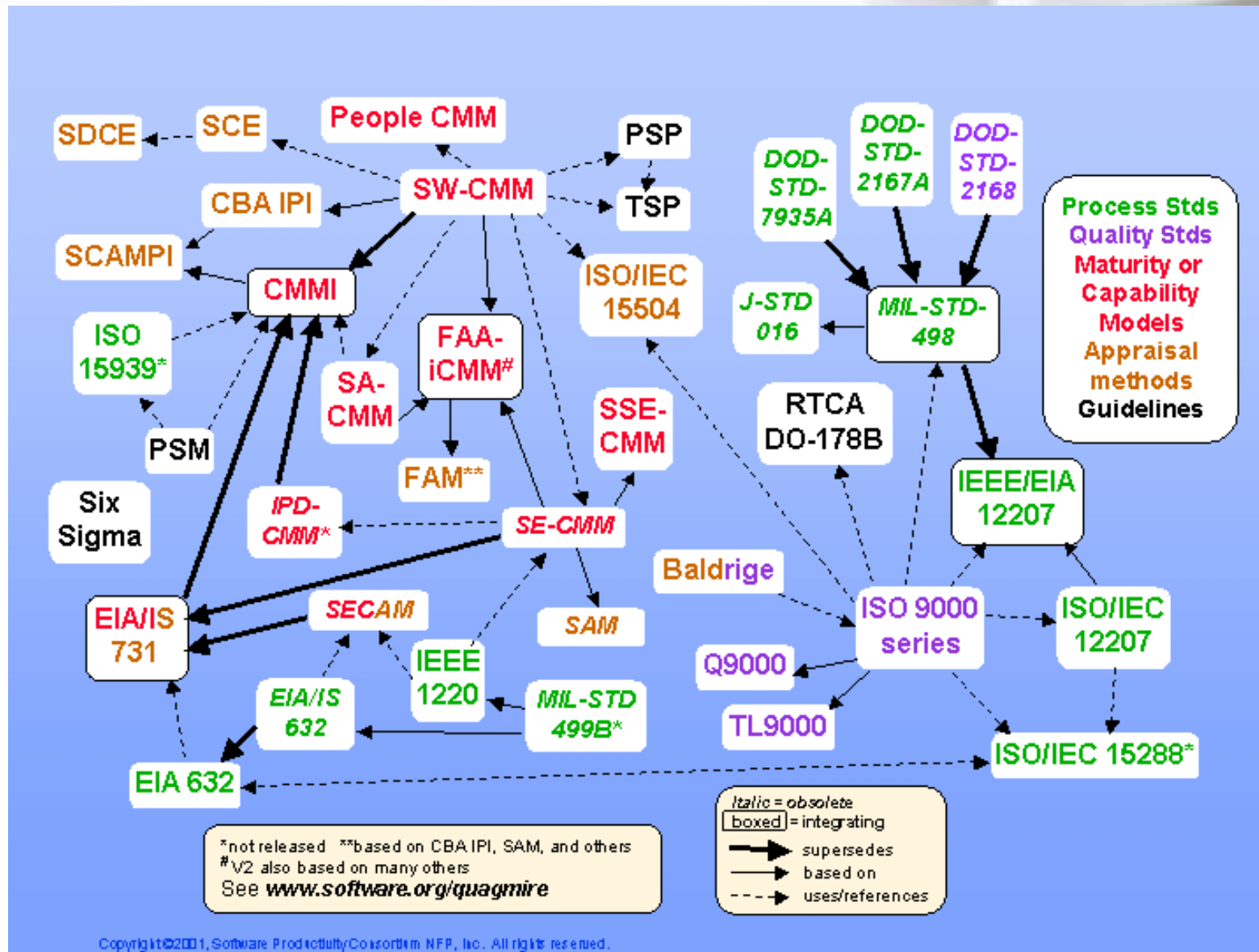
- **Avantages :**

- Rationalisation de l'économie,
- Construction de la qualité des biens et des services,
- Protection du consommateur....

- **Inconvénients :**

- Lourdeurs,
- Freins,
- Coûts....

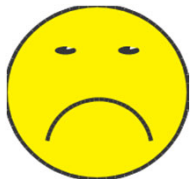
Référentiels Qualité



Les objectifs du cours



- Il s'agit de :
 - 1 D'assimiler et mettre en œuvre les concepts de l'assurance qualité logiciel
 - 2 Comprendre les grands concepts de 2 référentiels Qualité : ISO 9001 et CMMi



- Il ne s'agit pas de :
 - maîtriser les 2 référentiels ISO 9001 et CMMI

Assurance qualité logiciel

- ✓ **Les deux facettes de la qualité**
 - **Conformité avec la définition** (contrôlable en cours de fabrication ainsi qu'en maintenance)
 - **Réponse à l'attente du client** (contrôlable à la livraison principale ainsi que lors des livraisons intermédiaires)

Assurance qualité logiciel : **Contrôle Qualité**

- ✓ **Contrôle qualité**
- Activité tout au long du cycle de vie – plus de 50% des erreurs sont découvertes en phase d'exploitation
- le coût de réparation croît exponentiellement avec l'avancée dans le cycle de vie

Assurance qualité logiciel : Terminologie

✓ Terminologie

➤ Validation

. Faisons-nous le bon produit ?

➤ Vérification

. Faisons-nous le produit correctement ?

▣ *Attention, en pratique*

-) souvent confondus, ou pris l'un pour l'autre – on parle de « **V&V** » (validation et vérification)

Assurance qualité logiciel : Terminologie IEEE

- ✓ **IEEE** *Institute of Electrical and Electronics Engineers*
 - Promouvoir la connaissance dans les domaines de l'ingénierie électrotechnique et électronique.

- **Erreur :**

- commise par le développeur
- conduit à un défaut

- **Défaut :**

- imperfection dans le logiciel
- conduit ou non à une panne

- **Panne :**

- comportement anormal d'un programme Terme courant mais ambigu : bogue (bug)



Assurance qualité logiciel : Approche de MacCall

✓ **Caractéristiques « externes »**

- facteurs de qualité

✓ **Caractéristiques « internes »**

- critères de qualité

✓ **Caractéristiques mesurables**

- métriques

Approche de MacCall : **Facteurs de qualité**

- ✓ **Facteurs de qualité classés en 3 catégories :**
 - **Qualités opérationnelles**
(product operation)
 - **Facilité à changer ou à corriger**
(product revision)
 - **Facilité à faire des transitions d'environnement**
(product transition)

Facteurs de qualité (1): **Qualités opérationnelles**

- ✓ **Conformité aux besoins :**
 - Le produit fait-il ce qu'on souhaite/spécifications ?
- ✓ **Fiabilité :**
 - Le fait-il correctement dans tous les cas ?
- ✓ **Efficacité :**
 - Utilise-t-il au mieux les ressources matérielles / logicielles ?
- ✓ **Intégrité / Sécurité:**
 - Est-il protégé contre les intrusions ?
- ✓ **Exploitation / Utilisabilité /Facilité d'emploi :**
 - ... au niveau de l'apprentissage, de la mise en œuvre, de la fourniture des données, de l'interprétation des résultats, etc

Facteurs de qualité (2): **Facilité à changer ou à corriger**

✓ **Maintenabilité :**

- Facilité avec laquelle on peut localiser et corriger les erreurs

✓ **Flexibilité :**

- Facilité de modification et d'évolution

✓ **Testabilité :**

- Effort requis pour tester (après modification)

Facilité à faire des transitions d'environnement

✓ **Portabilité :**

- Peut-on utiliser le logiciel sur une autre machine ?

✓ **Réutilisabilité :**

- Peut-on réutiliser des parties du logiciel dans d'autres applications ?

✓ **Interopérabilité :**

- Facilité d'interfaçage avec d'autres systèmes

Influence des facteurs de qualité les uns sur les autres

✓ Ex. facteurs qui diminuent l'efficacité

- Intégrité (nécessité d'introduire des vérifications),
- Maintenabilité (sacrifice de l'efficacité pour la lisibilité),
- Portabilité (moins d'efficacité des structures portables).
- Testabilité, flexibilité, réutilisabilité, interopérabilité,.....

✓ Ex. facteurs qui diminuent l'intégrité

- Flexibilité,
- Réutilisabilité,
- Interopérabilité.

Critères de qualité

Opérabilité

Apprentissage

Volume et taux d'E/S

Communicabilité

Contrôle d'accès

Consommation mémoire

Vitesse d'exécution

Traçabilité

Complétude

Précision

Cohérence

Tolérance aux fautes

Simplicité

Modularité

Concision Auto-description

Instrumentation

Généralité

Evolutivité

Indépendance machine

Indépendance système

Communications banalisés

Données banalisées

Ex : Trouver les (>40) liens entre facteurs de qualité et critères de qualité

Critères de qualité
(caractéristiques internes)

Opérabilité
Apprentissage
Volume et taux d'E/S
Communicabilité
Contrôle d'accès
Consommation mémoire
Vitesse d'exécution
Traçabilité
Complétude
Cohérence
Tolérance aux fautes
Simplicité
Précision
Modularité
Concision
Auto-description
Instrumentation
Généralité
Évolutivité
Indépendance machine
Indépendance système
Communications
banalisés
Données banalisées



Facilité d'emploi
Intégrité
Efficacité
Conformité
Fiabilité
Maintenabilité
Testabilité
Flexibilité
Portabilité
Réutilisabilité
Interopérabilité

Facteurs de qualité
(caractéristiques externes)

Confert solution proposée

✓ Confert solution proposée



M03_2021_Solution_Exercice_Assurance_Qualité.pptx - PowerPoint

Métriques associées aux critères de qualité

✓ **Mesure des caractéristiques « internes »**

➤ Mesures objectives

- Taille, complexité du flot de contrôle, cohésion modulaire / couplage entre modules, ...

✓ **Mesure des caractéristiques « externes »**

➤ Evaluations stochastiques (statistiques)

- ex. : délai moyen de réponse à une requête, nombre de requêtes simultanées sans « écrouler » un serveur, ...

Métriques associées aux critères de qualité

✓ **Mesure des caractéristiques « internes »**

➤ Mesures objectives

- Taille, complexité du flot de contrôle, cohésion modulaire / couplage entre modules, ...

✓ **Mesure des caractéristiques « externes »**

➤ Evaluations stochastiques (statistiques)

- ex. : délai moyen de réponse à une requête, nombre de requêtes simultanées sans « écrouler » un serveur, ...

☞ Ce sont des mesures a posteriori

→ arrivent parfois « trop tard »

Exemples de mesures de critères de qualité (1)

- **Fiabilité :**
 - mesures stochastiques :
 - temps moyen de réparation
 - temps moyen entre deux pannes
 - taux de disponibilité
- **Portabilité :**
 - mesure objective :
 - nb d'instructions dépendant de la plate-forme cible

Exemples de mesures de critères de qualité (2)

- Facilité d'utilisation :
 - mesures objectives :
 - nb de paramètres ayant une valeur par défaut (pertinente)
 - nb d'écrans d'aide
 - mesures stochastiques :
 - nb de fausses manipulations par jour
 - nb de jours d'apprentissage
 - temps de lecture / interprétation des résultats affichés

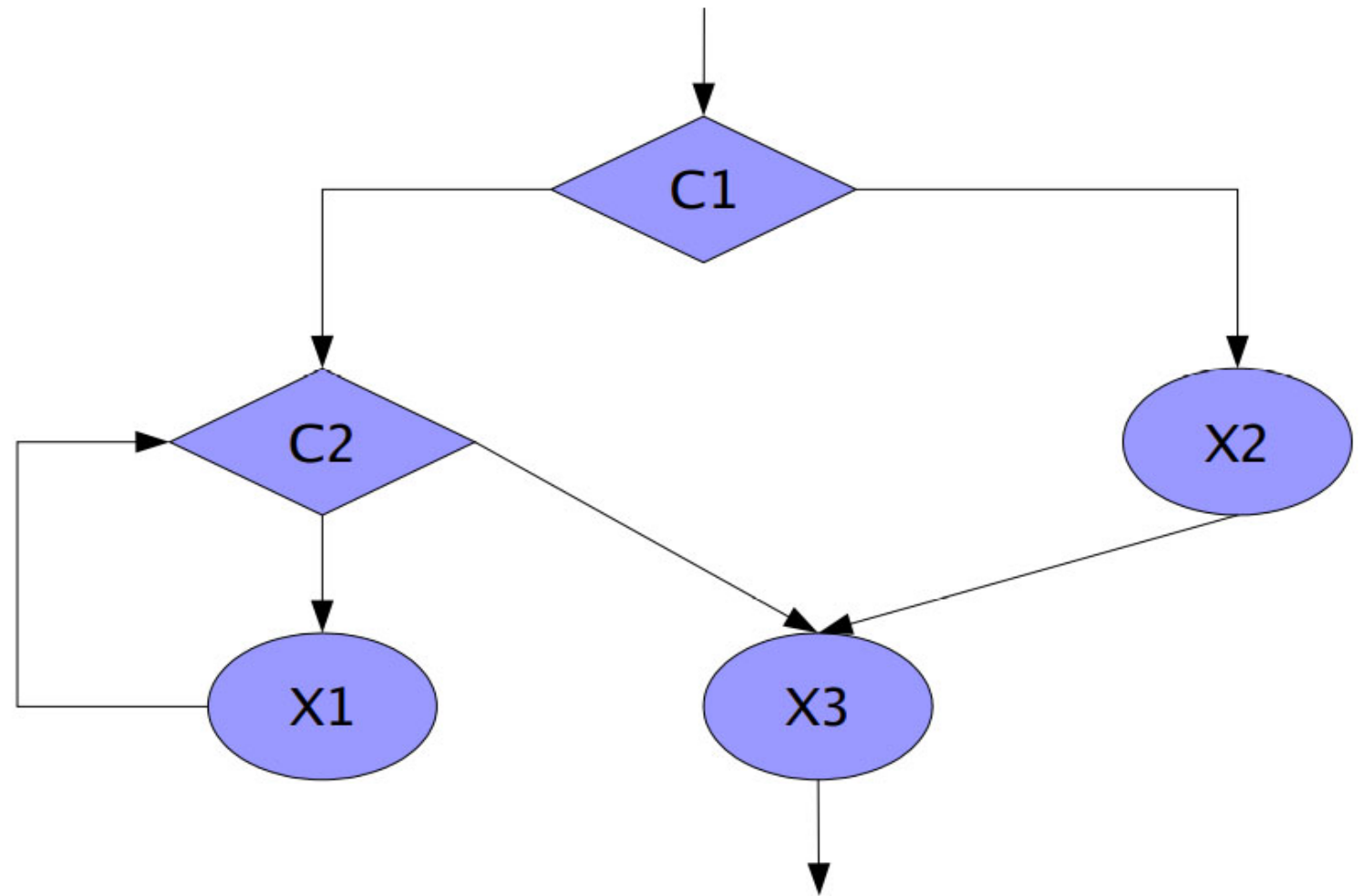
Métriques

- Métriques
 - de Halstead, de McCabe, de Henry et Kafura, ..
- Quantités mesurées
 - concision (taille programme / taille de l'algorithme), complexité textuelle, difficulté, effort, complexité des liaisons inter-modules (ou -classes), encombrement d'une classe, complexité structurelle, ...
- Implémentées par des outils
 - Logiscope, ...

Exemple : Métrique de McCabe

- Analyse du graphe de contrôle
- Mesure de la complexité structurelle
- **Nombre cyclomatique**
 - = nb de noeuds (blocs d'instructions séquentielles)
 - nb d'arcs (branches de programme)
 - + nb points d'entrée
 - + nb points de sortie
- Représente le nombre de chemins indépendants
 - ~ nb conditions + 1 (si les décisions sont binaires)

Exemple : Métrique de McCabe



nb entrées = 1
nb sorties = 1
nb noeuds (blocs) = 5
nb arcs (branches) = 6
nb cyclomatique = $6 - 5 + 1 + 1 = 3$
(nb décisions = 2)

QUALITE : Comment faire ?

Un grand nombre de métriques

- difficile de les connaître toutes, et souvent discutables (vision partielle/artificielle du problème)

L'important, c'est la démarche :

- définir un plan qualité **adapté au contexte**
- détailler une mesure des différents critères
- s'interroger sur la validité des métriques (pertinence)

Problème des mesures a posteriori

- trop tard, mais forge l'expérience (😊)

Assurance qualité et gestion de projet (1)

Le contrôle qualité aide la gestion projet :

- mesure de l'avancement
- meilleure estimation des coûts

Assurance qualité et gestion de projet (2)

Mais (écueils) :

- dilemme : moins cher ou plus sûr ?
 - souvent prééminence du planning sur la qualité
- sous-estimation des ressources qui sont nécessaires à l'assurance qualité
 - par les développeurs (activité jugée marginale)
 - par les managers (réticence à prévoir un budget maintenance)

Moyens de l'assurance qualité ?

- Méthodes statiques a priori
(sans exécuter le logiciel)
 - examen critique de documents et de code
 - analyse automatique (ou assistée) de code / spécification
 - analyse statique de programme
 - vérification de modèle (model checking)
 - outils de preuves formelles
- Méthodes dynamiques a posteriori
(en exécutant le logiciel)
 - test

Moyens de l'assurance qualité ?

- Méthodes statiques a priori
(sans exécuter le logiciel)
 - examen critique de documents et de code
 - analyse automatique (ou assistée) de code / spécification
 - analyse statique de programme
 - vérification de modèle (model checking)
 - outils de preuves formelles
- Méthodes dynamiques a posteriori
(en exécutant le logiciel)
 - test

Examen critique de documents : Revue par les pairs (1)

- Avoir un point de vue différent de l'auteur
 - quelqu'un d'autre
- Avoir différents points de vue
 - compétences multiples
- Avoir des points de vue objectifs
 - participants hors de l'équipe de développement

Examen critique de documents : Revue par les pairs (2)

- Critiquer les aspects techniques, pas l'auteur (!)
- Juger la forme
 - format [voir cours sur la documentation], structure, satisfaction des normes du plan qualité...
- Juger le fond
 - précision, non-ambiguïté, complétude, cohérence (pas de référence imprécise ou inexistante)
 - conformité par rapport aux documents amont, au plan projet

Coûts des inspections (ordre de grandeur)....

- Cahier des charges 5-10 pages/h
- Spécifications fonctionnelles 10 pages/h
- Conception globale 5-15 pages/h
- Conception détaillé 10 pages/h
- Code 20-50 lignes/h

(Selon moi :

- effort trop faible dans les phases amont
- effort difficile et peu productif dans les phases aval)

Méthodes de relecture (1)

- Auto-correction (desk-checking)
 - relecture personnelle
 - bilan : efficacité quasi nulle pour les documents amont, faible pour le code
- Lecture croisée (author-reader cycle)
 - un collègue recherche des ambiguïtés, oublis, imprécisions
 - bilan : efficacité faible pour les documents amont, plus adapté pour la relecture du code

Méthodes de relecture (1)

- Revue (walkthrough)
 - discussion informelle au sein d'un groupe
 - un lecteur résume paragraphe par paragraphe
 - bilan : contribution moyenne à l'assurance qualité (évaluation très liée à la prestation du lecteur)
- Revue structurée
 - constitution pendant le débat d'une liste de défauts, utilisation d'une liste de défauts typiques (checklist)
 - direction des débats par un secrétaire
 - bilan : bonne contribution à l'assurance qualité

Méthodes de relecture (2)



- Inspection
 - cadre de relecture plus formel
 - recherche des défauts avant les débats
 - suivi des décisions et corrections
 - bilan : excellente contribution à l'assurance qualité

Méthodes de relecture : Inspection (1)

Organisation

1. préparation

- recherche des défauts
- rédaction d'un rapport de défauts basé sur des fiches type

2. cycle de réunions

- examen des défauts
- prise de décision

3. suivi

- vérification des corrections ou nouvelle inspection

Méthodes de relecture : Inspection (2)

Planification – pour chaque type de document :

- dates de début et de fin par rapport au plan projet
- critères de sélection des inspecteurs
- plan d'inspection (parties à inspecter)
- types de défauts les plus communs (checklist)
- formulaires d'inspection (description de défauts)
- critères de succès de l'inspection

(Ce plan figure en partie dans le plan qualité)

Responsabilités

- inspecteurs :
 - responsables de la qualité du produit final
 - responsables du respect des principes de qualité
- auteur :
 - mise à disposition des documents à la date prévue
 - fournit une opinion sur chaque défaut signalé
- ...

Méthodes de relecture : Inspection (4)

Responsabilités

- ...
- secrétaire :
 - enregistre les défauts considérés et les décisions prises
 - assiste le modérateur
- modérateur :
 - responsable du bon déroulement des réunions
 - convocation, tenue, suspension
 - veille au maintien des objectifs et aux facteurs humains
 - préside la prise de décision

Méthodes de relecture : Inspection (5)

- Conséquence
 - la formalisation oblige à planifier et à observer les principes de qualité
- Bilan
 - excellente contribution à l'assurance qualité
 - amélioration du cycle de vie (contrôle au plus tôt)
 - influence positive sur la communication et la formation dans le projet

👉 la meilleure des méthodes de relecture

Méth. dynamiques a posteriori (en exécutant le logiciel) : TESTS

- Vérification (conformité aux spécifications)
 - tests unitaires
 - tests d'intégration
- Qualification
 - validation par rapport aux contraintes non fonctionnelles
 - tests de performance
 - tests de capacité de charge
 - validation par rapport aux besoins
 - bêta-test (chez l'utilisateur final)

SYNTHESE : A RETENIR.....

- Contrôle qualité : durant tout le cycle de vie
- L'important, c'est la démarche
 - définir un plan qualité adapté au contexte
 - détailler les mesures des critères (a posteriori ☹)
 - s'interroger sur la pertinence des métriques
- Méthodes statiques (sans exécuter) :
 - inspection : par des extérieurs, checklist de défauts
 - analyse automatique (ou assistée) de programme
- Méthodes dynamiques (en exécutant) : test

Méth. dynamiques a posteriori (en exécutant le logiciel) : TESTS

- Vérification (conformité aux spécifications)
 - tests unitaires
 - tests d'intégration
- Qualification
 - validation par rapport aux contraintes non fonctionnelles
 - tests de performance
 - tests de capacité de charge
 - validation par rapport aux besoins
 - bêta-test (chez l'utilisateur final)

Et au stade de la recherche avancée.....

- Génération automatique de programmes à partir des spécifications
 - programmes corrects par construction
- Mais
 - méthode encore trop mathématique pour le public des développeurs
 - problème d'efficacité du code généré
 - problème de rendement (malgré l'automatisation)
 - beaucoup de choses à spécifier avec soin
 - dilemme bien connu : moins cher ou plus sûr ?

Normes ISO 9000

- Ensemble de normes internationales (International Standard Organisation) de gestion de la qualité depuis 1987
- Révisions tous les 5 ans en moyenne
- Applicable à plusieurs domaines (manufacturier, service,...) et à toute l'entreprise (organisme)
- Satisfaire à la norme ISO 9000 démontre la capacité d'une organisation à produire des biens et services en répondant aux exigences clients et réglementaires
- Certification par un organisme indépendant (ex: AFAQ)
- Référencement de la certification
- Entretien régulier :
 - audit de suivi tous les ans pendant 3 ans puis audit de renouvellement tous les 3 ans
 - Audits internes

Les objectifs ISO 9001:2015

Mettre en œuvre un système de management de la qualité :

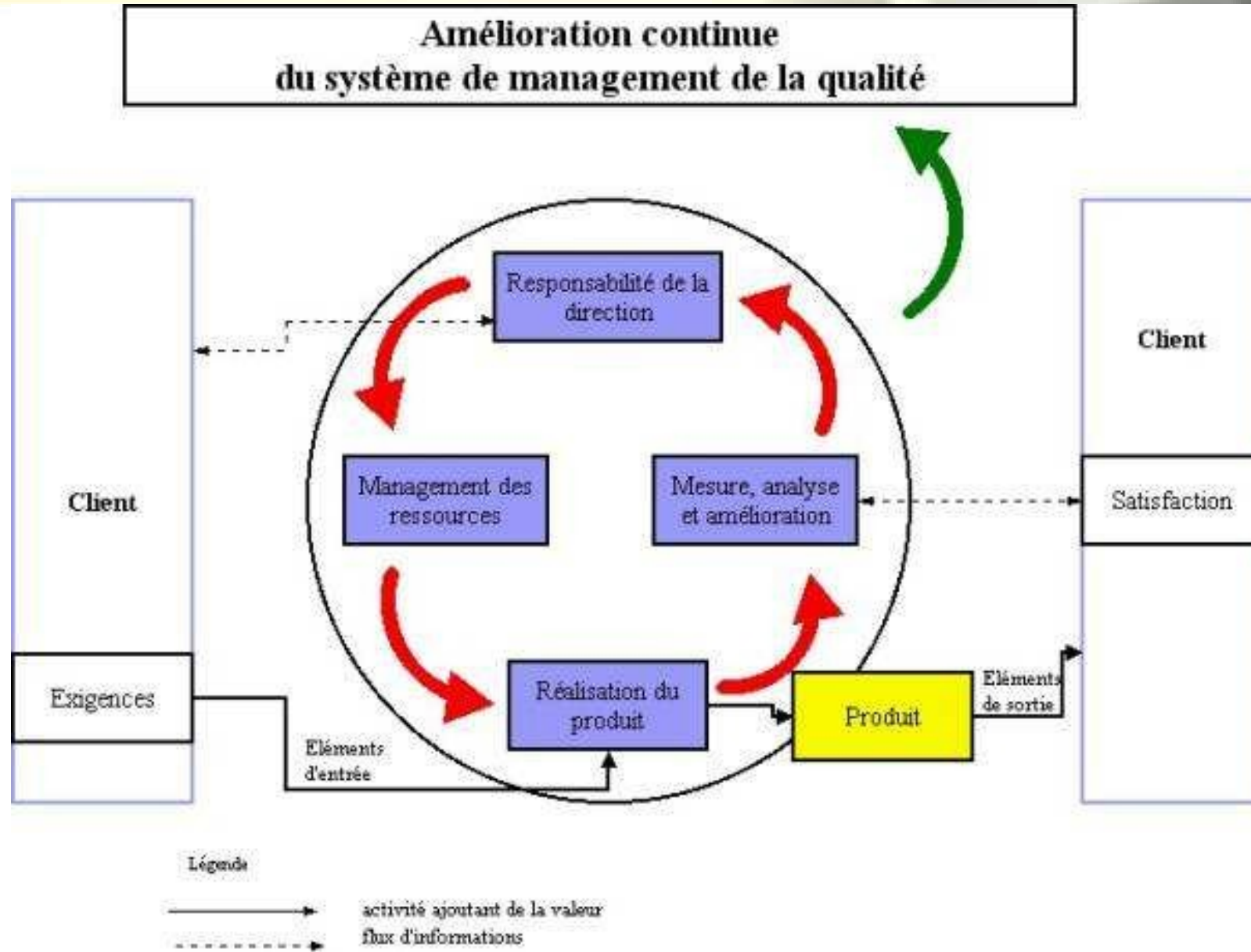
- Fournir **régulièrement** un produit conforme aux **besoins** du client et aux exigences réglementaires applicables
- Chercher à **accroître la satisfaction des clients** par l'application efficace du système et mettre en œuvre un **processus d'amélioration continue**.

Les principes ISO 90001:2015

8 Principes :

1. Approche processus
2. Ecoute client
3. Leadership
4. Implication du personnel
5. Management par approche système
6. Amélioration continue
7. Approche factuelle pour la prise de décision
8. Relations mutuellement bénéfiques avec les fournisseurs

Les processus clés de ISO 9001:2015



Les chapitres ISO 9001:2015

- 0 - Introduction
- 1 - Domaine d'application
- 2 - Références normatives
- 3 - Termes et définitions
- 4 - *Système de Management de la Qualité*
- 5 - *Responsabilité de la direction*
- 6 - Management des ressources
- 7 - Réalisation du produit
- 8 - Mesures, analyses et amélioration

Chapitre 4

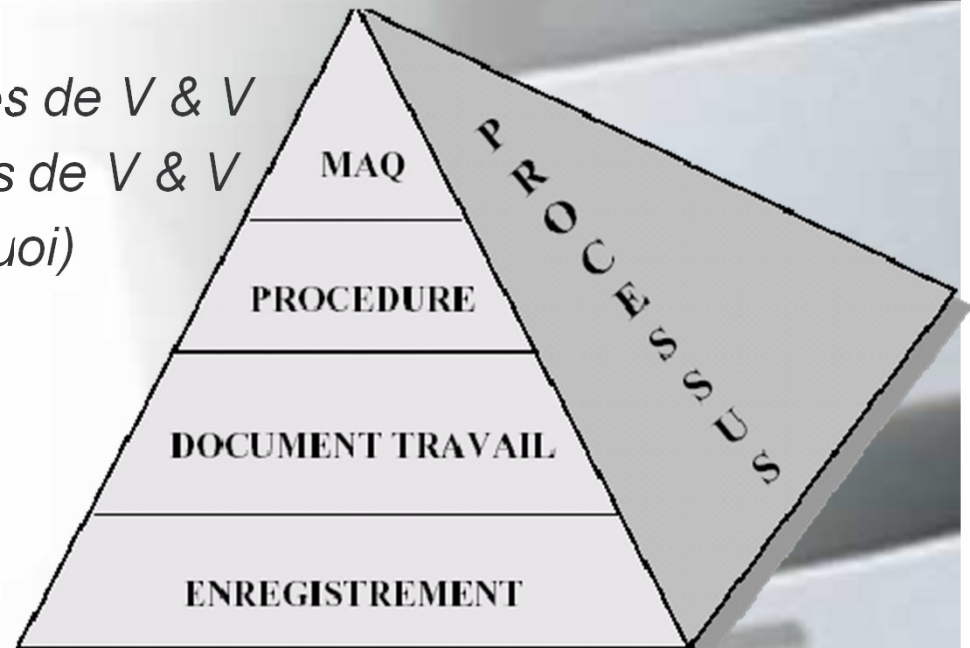
- **Systeme de Management de la Qualite**

- Architecture documentaire

- *Manuel qualite*

- *Plan qualite*

- *objectifs de la qualite (en termes mesurables)*
- *criteres d'entree et de sorties de chaque phase*
- *identification des activites de V & V*
- *planification des activites de V & V*
- *responsabilite (qui fait quoi)*
- *Criteres d'acceptabilites*
- *Enregistrements*



Mise en oeuvre ISO 9001:2015

1. **Mise en place d'un SMQ** et du MAQ associé + formation du personnel à l'utilisation (avec un consultant spécialisé)
2. **Audit "à blanc"** par un auditeur expérimenté
3. **Certification** par un organisme accrédité, la certification est prononcée si l'auditeur ne trouve pas de non conformité sinon après correction des dysfonctionnement un nouvel audit a lieu

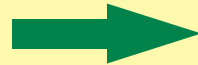
Le CMMi

- Le **Capability Maturity Model Integration** est un modèle Qualité conçu par le Software Engineering Institute (SEI) pour le ministère de la Défense du gouvernement Américain (DoD)
- Il permet d'évaluer et de faire évoluer la maturité d'une organisation à produire du logiciel
- Certification par des évaluateurs accrédités par le SEI (Software Engineering Institute)
- Référencement de la certification et d'entretien obligatoire pour conserver le référencement

Qu'est-ce que le CMMi ?

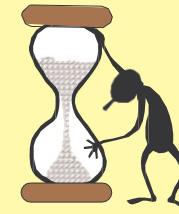
- Un modèle Qualité conçu pour améliorer le processus de développement et de maintenance dans le but :

- d'améliorer la qualité des produits développés



Satisfaction des clients

- de réduire le 'délai mise sur le marché'



- d'optimiser les efforts :

- de développement
- d'intégration
- de validation



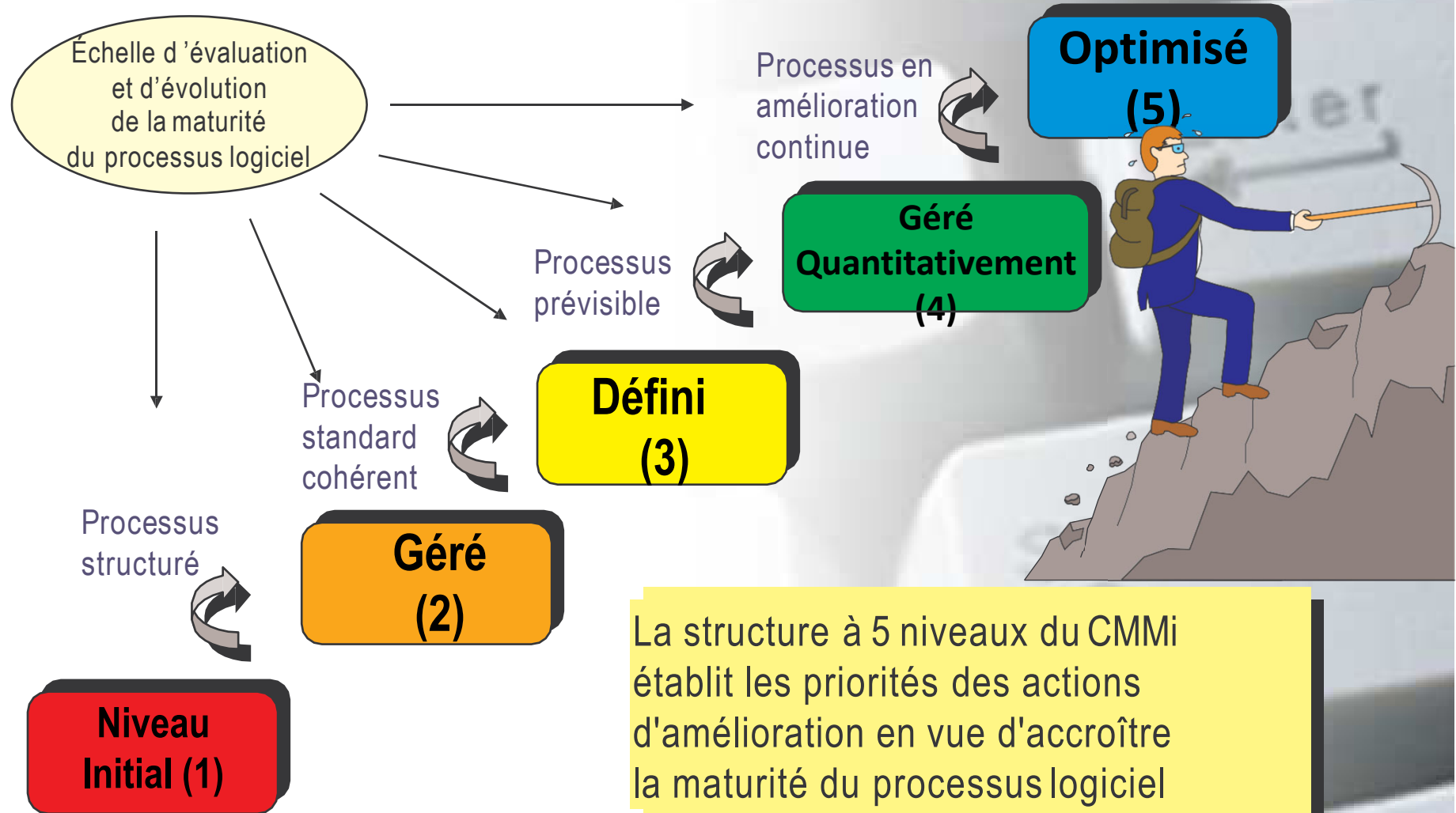
Moins de 're-travail'



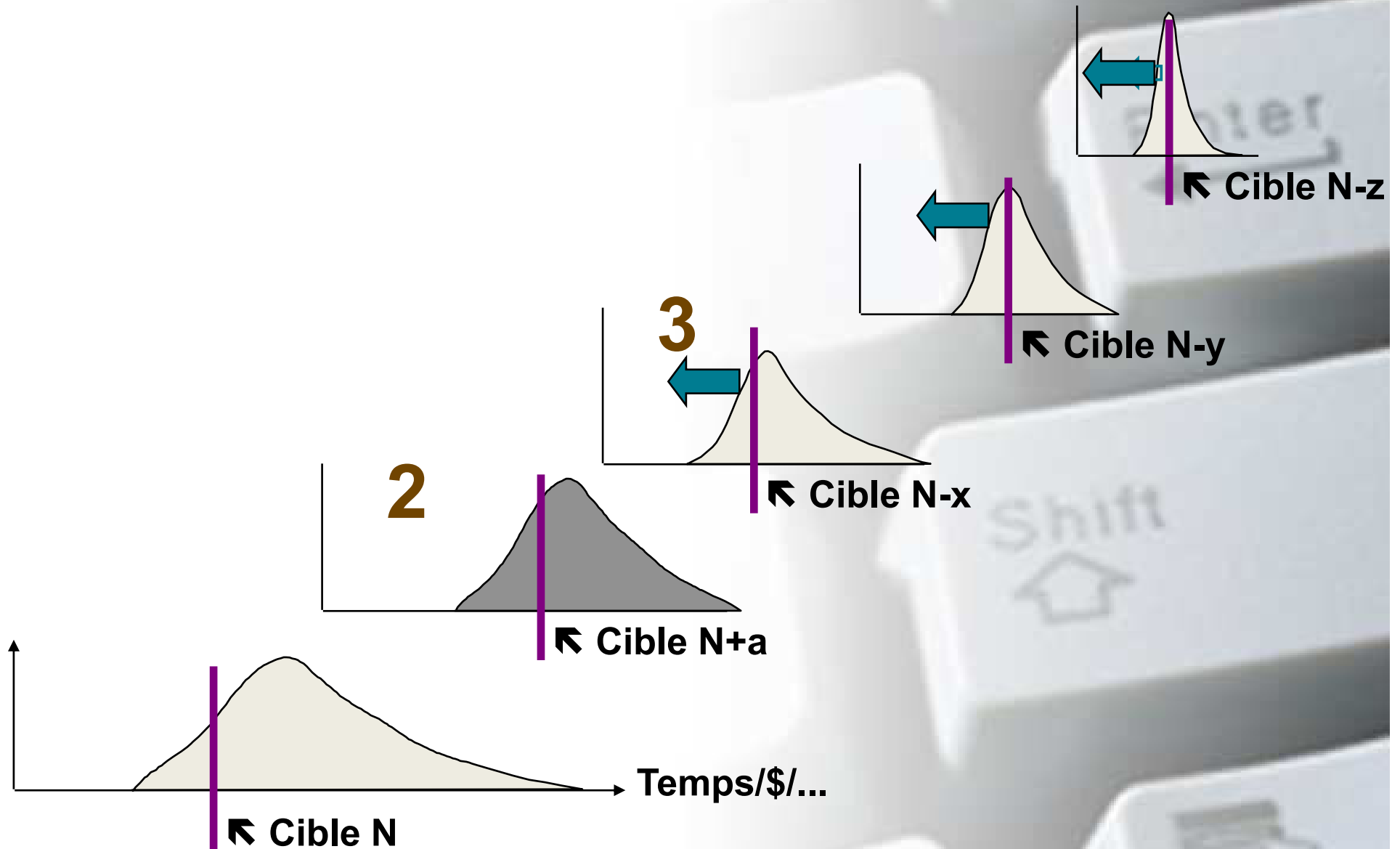
“ La qualité d'un système logiciel est régie par la qualité du processus utilisé pour le développer et le faire évoluer. ”

Watts S. Humphrey (concepteur du modèle CMM)

Les 5 niveaux de maturité du processus logiciel



La capacité d'atteindre les objectifs (coûts, délais, qualité) en fonction du niveau de maturité



Les 5 niveaux de maturité du processus logiciel : N1

- **Initial** (initial, niveau de maturité 1): La réussite des projets dépend du savoir-faire de quelques personnes clés dans l'organisation, pas de formalisation des processus et pas de partage.
- Ce niveau est caractérisé par un ensemble de mauvaises pratiques et d'un seul non-processus, le mode « *pompier* » avec une gestion par crise:
 - Indiscipline généralisée
 - Pas de processus fiables, on s'en remet à l'expérience de la personne
 - Projets pilotés par les délais
 - Population de héros
 - Pas d'enseignement tiré des difficultés ou erreurs
 - Pas de contrôle
 - Mode essentiellement réactif
 - Incapacité à reproduire les éventuels succès passés

Les 5 niveaux de maturité du processus logiciel : N2

- **Managed** (géré, niveau de maturité 2): une gestion de projet élémentaire est définie pour assurer le suivi des coûts, des délais et de la fonctionnalité du projet. La discipline nécessaire au processus est en place. Le chef de projet a une forte responsabilité : il doit définir, documenter, appliquer et maintenir à jour ses plans. D'un projet à l'autre, il capitalise et améliore ses pratiques de gestion de projet et d'ingénierie. Mais il ne s'interroge pas sur la pertinence de l'expression de besoin et il se limite à sa réalisation. Les 7 processus élémentaire doivent être maîtrisés:
 - Gestion des exigences
 - Planification du projet
 - Conduite et maîtrise du projet
 - Gestion des achats
 - Production et analyses des indicateurs
 - Assurance qualité des processus et des produits
 - Gestion de la configuration

Les 5 niveaux de maturité du processus logiciel : N3

- **Defined** (défini, niveau de maturité 3): Ce niveau est caractérisé par une standardisation adéquate des pratiques, une capitalisation centralisée (en particulier sur les mesures réalisées dans les projets) et une maîtrise du référentiel interne (ou **Systeme de management de la qualité**).
- L'organisation a défini des méthodes, outils et document.
- L'organisation est garante de la pérennité de l'ensemble, le chef de projet n'est plus tout seul. Il existe des lignes directrices, un plan stratégique et une planification de l'amélioration de processus pour le futur, en ligne avec les objectifs de l'organisation. Les employés sont formés et conscients de leurs responsabilités ainsi que de leurs devoirs.

Les 5 niveaux de maturité du processus logiciel: N4

- **Quantitatively managed** (quantifié, niveau de maturité 4): Les projets sont pilotés sur la base d'objectifs quantitatifs.
- On travaille sur la notion de productivité, de performance des projets, en récupérant et comparant les statistiques de chaque projet.
- Des mesures détaillées sont prises en ce qui concerne le déroulement du processus logiciel et la qualité des produits.
 - Métriques / Indicateurs mis en place et exploités
 - Retours d'expérience possibles car processus cohérents (les comparaisons ont un sens)
 - Programme qualité
 - Évaluation des impacts liés aux évolutions de processus

Les 5 niveaux de maturité du processus logiciel: N5

- **Optimizing** (optimisé, niveau de maturité 5): Les processus qui sont gérés quantitativement pour le pilotage de projet sont en optimisation constante par l'organisation afin d'anticiper les évolutions prévues (besoins clients, nouvelles technologies...).
- Les statistiques sont utilisées pour les processus standards.
- Le référentiel sert à construire et améliorer les processus standards

Chaque niveau intègre un certain nombre de processus qui doivent être maîtrisés par l'organisation ou par le projet.

Les 5 niveaux de maturité de CMMi permettent une amélioration continue par étapes

Les secteurs constituent une façon de décrire comment l'organisation gagne en maturité

Niveau	Libellé	Secteur	
5	Processus en amélioration continue	<ul style="list-style-type: none"> Gestion et déploiement des innovations au niveau organisation (OID) Prevention de défaut (analyses Causales) (CAR) 	
4	Processus Géré quantitativement	<ul style="list-style-type: none"> Performance du processus de l'organisation (OPP) Gestion de projet Quantitative (QPM) 	
3	Processus Défini (standardisé)	<ul style="list-style-type: none"> Développement des exigences (RD) Conception de la solution technique (TS) Intégration des Produits (PI) Verification (VER) Validation (VAL) Focalisation processus de l'organisation (OPF) Definition du processus de l'organisation (OPD) Plan de formation de l' organisation (OT) Gestion intégrée de Projet (IPM) Gestion des Risques (RSKM) Equipes Intégrée (IT) Gestion intégrée des fournisseurs (ISM) Prise de décision et Resolution (DAR) Environnement organisationnel pour l' Integration (OEI) 	
2	Processus Géré : Gestion de projet efficace	<ul style="list-style-type: none"> Gestion des exigences (REQM) Planification de projet (PP) Suivi et supervision de projet (PMC) Gestion contractuelle de fournisseur (SAM) Mesures et Analyses (M&A) Assurance de la Qualité du processus et des Produits (PPQA) Gestion de Configuration (CM) 	
1			

Le niveau 2 du CMMi

- Niveau Géré :
 - Pratiques saines de gestion de projet en place
 - Gestion des nouveaux projets est fondée sur l'expérience acquise
- 7 Secteurs clés :
 - Gestion des exigences
 - Planification de projet logiciel
 - Suivi et supervision de projet logiciel
 - Gestion de la sous-traitance logiciel
 - Mesures et analyses
 - Assurance-qualité logiciel
 - Gestion de la configuration logiciel

Les principaux rôles de CMMi

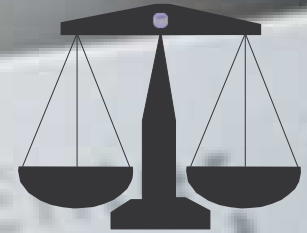
- **Direction** : gestion de l'organisation avec comme objectif la vitalité long terme de l'organisation.
- **Chef de projet** : responsable de l'ensemble du projet (logiciel, matériel,): responsable ultime vis-à-vis du client
- **Personnel d'ingénierie Logiciel** : personnel technique (analystes, programmeurs, testeurs,....) : responsable de tâches logiciel ou chargé d'accomplir une fonction donnée (gestion de configuration logiciel...)
- **Groupe Assurance Qualité** : indépendant du développement, il fournit une vue objective des progrès en termes de produits et de suivi du processus.
- **Groupe Gestion de Configuration Logiciel** : responsable de la coordination et de la mise en œuvre CGL pour le projet.
- **Groupe Amélioration des Processus** (Software Engineering Process Group) : responsable développement et amélioration du processus Logiciel

Démarche de mise en œuvre du CMMi

- **Définition des objectifs**
- **Audit interne (Gap Analysis)**
- **Identification des secteurs prioritaires**
- **Mise en place des processus**
- **Audit à blanc**
- **Evaluation (assessment)**
- **Bilan**

Coûts/Bénéfices de CMMi (sources SEI)

- C'est long et coûteux :
 - **5 % en moyenne du budget global d'une organisation** (moins cher pour les grosses organisations)
 - **Minimum 18 mois pour passer au niveau 2**
- Mais on gagne ensuite beaucoup : ROI admis 5 pour 1 si appliqué de façon consistante pendant 3 à 5 ans
 - **+ 35% de productivité par an**
 - **délai amélioré de 19%**
 - **+ 22% de défauts trouvés en amont**
- Quels sont les éléments permettant de faire de tels gains ?
 - **Management de projet très rigoureux** + gestion des besoins client et surtout des changements, sous le contrôle d'une organisation QA efficace
 - **Processus de revue par les pairs** (Peer review) très strict :
 - ✓ - de : défauts, de temps de tests, coûts, temps de développement, de maintenance



ISO / CMMi : Synthèse

	ISO 9001	CMMi
Périmètre	Système qualité de toute l'entreprise + satisfaction client	Possibilité de cibler sur activité informatique ou toute l'entreprise
Mise en œuvre	+ Principes généraux	+ précis sur pratiques (guide) + progressive (niveaux)
Accréditation	Formelle, entretien => coûteuse mais fiable	Formelle, entretien => très stricte (beaucoup de preuves !)

Tendance :

- le CMMi devient aujourd'hui **la norme de facto** sur l'amélioration des processus dans le domaine du logiciel

Ex 1: QUIZ ISO9001



- 1- Donnez une définition de la Qualité
- 2- Qu'est-ce que la norme internationale ISO 9001 ?
- 3- Selon vous , la Qualité dépend principalement :
 - du Responsable du Service Qualité?
 - du Responsable de l'entreprise (Organisme)?
 - de tout le Personnel de l'entreprise (Organisme)?
- 4 - L'Assurance Qualité est ...
 - la garantie pour un client d'être toujours satisfait?
 - une démarche applicable principalement dans le monde industriel?
 - l'assurance, pour un Client, d'un dédommagement financier en cas de non conformité d'un produit ou service acheté?
- 5 - La Certification Qualité ISO-9001 ...
 - est la seule façon de prouver la qualité de son offre vis à vis de l'extérieur?
 - garantit essentiellement que l'organisme est capable de répondre aux besoins actuels et futurs de ses clients?
 - la dernière étape facultative de mise en place d'un système de management de la qualité?

Ex 2: QUIZ CMMI



- 1- Qu'est ce que le CMMI ?
- 2- Combien existe –t-il de modèles disponibles pour les organisations ?
- 3- Quels sont les niveaux de maturité du modèle CMMI . Donner les spécificités et caractéristiques de chaque niveau ?
- 4 – Que représente le niveau de maturité dans le modèle CMMI ?
- 5 – Expliquez ce qu'est le niveau de maturité CMMI élevé ? Que représente t-il ?
- 6 – Qu'est ce qu'un secteur clé ? Donner 3 secteurs clés du niveau de maturité CMMI 2 ?

Confert solutions proposées



M03_2021_Solution_Exercice_Qualité_ISO9001_CMMI.pptx



Normalisation des processus
JP ROBERTI Janv 2022