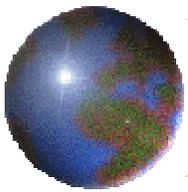


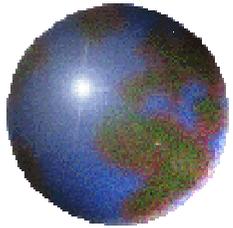
Qualité des systèmes d'information

Isabelle WATTIAU

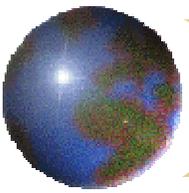


Sommaire

- ❖ 1. Introduction
- ❖ 2. Définitions, concepts et outils
- ❖ 3. Qualité des systèmes d'information
- ❖ 4. Qualité et normalisation
- ❖ 5. Qualité du logiciel
- ❖ 6. Qualité du processus logiciel
- ❖ 7. Conclusion

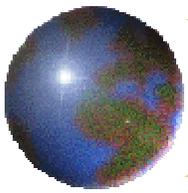


1. Introduction



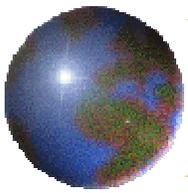
Introduction

- ❖ La qualité est un concept « à la mode » dans le domaine des Systèmes d'Information
 - ❖ Signe de maturité du domaine
- ❖ Première période : recherche d'efficacité
 - ❖ Le SI doit répondre aux besoins fonctionnels
- ❖ Deuxième période : recherche de qualité
 - ❖ Le SI doit répondre aussi aux besoins non fonctionnels
 - ❖ Qualité de service



Les enjeux

- ✚ Un SI de qualité est un actif valorisable
- ✚ Un SI de qualité peut améliorer la satisfaction du client
- ✚ Un SI de qualité peut procurer un avantage compétitif



Un exemple de problématique de qualité : le système d'information hospitalier

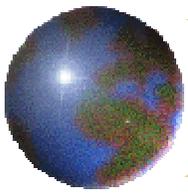
- ❑ **Objectif qualité** : mesurer la qualité du service rendu
 - ❑ de la traçabilité de l'information à la complétude dans les dossiers médicaux des patients

- ❑ **Approches qualité traditionnelles**
 - ❑ Tirage au sort de 80 dossiers
 - ❑ Évaluation manuelle

- ❑ **Points faibles** :
 - ❑ Lourd à mettre en œuvre
 - ❑ Etude rétrospective => Pas de possibilité de correction

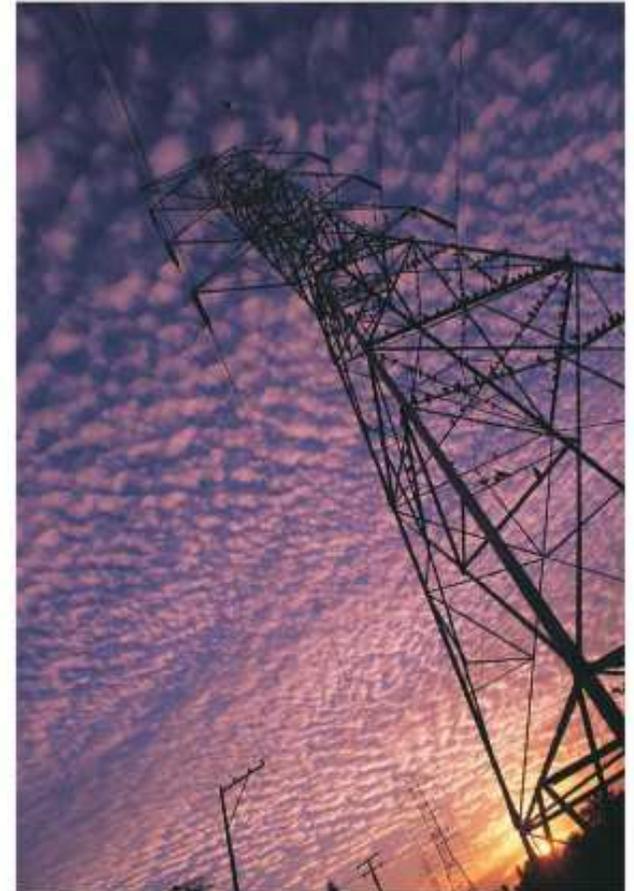
- ❑ **Les enjeux** :
 - ❑ Sécurité
 - ❑ Image
 - ❑ Coûts

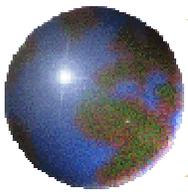




Un exemple de problématique de qualité : la consommation d'électricité

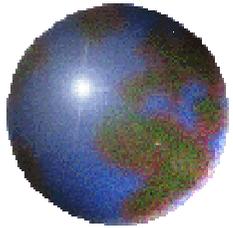
- ❑ **Objectif qualité** : Analyser les prévisions de consommation de l'électricité pour diverses catégories de consommateurs
 - ❑ nécessite une complétude et une cohérence de l'information
- ❑ **Questions clés** :
 - ❑ est ce que les données couvrent toutes les populations souhaitées ?
 - ❑ est ce que les données sont cohérentes ?
- ❑ **Exemples d'approches** :
 - ❑ Mesure du taux de données manquantes
 - ❑ Contraintes d'intégrité



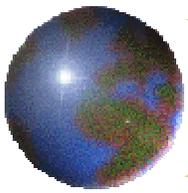


Les conséquences de la non-qualité

- ❁ Coûts : des implications simples
 - ❁ Qualité du logiciel -> coût de la maintenance
 - ❁ Non-qualité des données : 600 milliards de dollars par an aux USA (source : Data Warehousing Institute, 2002)
 - ❁ Dépendance vitale : l'explosion de la navette Challenger était due à des problèmes de qualité
 - ❁ « Bug » de l'an 2000 : 300 à 600 milliards de dollars
 - ❁ Etc.

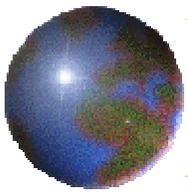


2. Définitions, concepts et outils



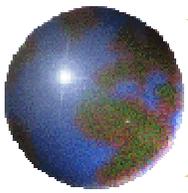
Définitions de la qualité

- ❖ 1. Toutes les actions que l'on peut entreprendre:
 - ❖ En amont, en cours de réalisation d'un projet pour améliorer le produit ou le service
 - ❖ Et en aval pour suivre le produit ou le service livré.
- ❖ 2. ISO8402:1994
 - ❖ L'ensemble des propriétés et caractéristiques d'un produit ou d'un service qui lui confère l'aptitude à satisfaire des besoins exprimés ou implicites
 - ❖ C'est donc une notion relative, fondée sur le besoin
 - ❖ On doit rechercher une qualité optimale et pas nécessairement une qualité maximale



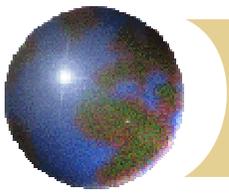
Typologie de la qualité

- ⊕ Qualité attendue : celle que le client souhaite
- ⊕ Qualité voulue : que le fournisseur a prévu de faire
- ⊕ Qualité réalisée : traduit ce qui a été obtenu
- ⊕ Qualité perçue : ce que le client constate

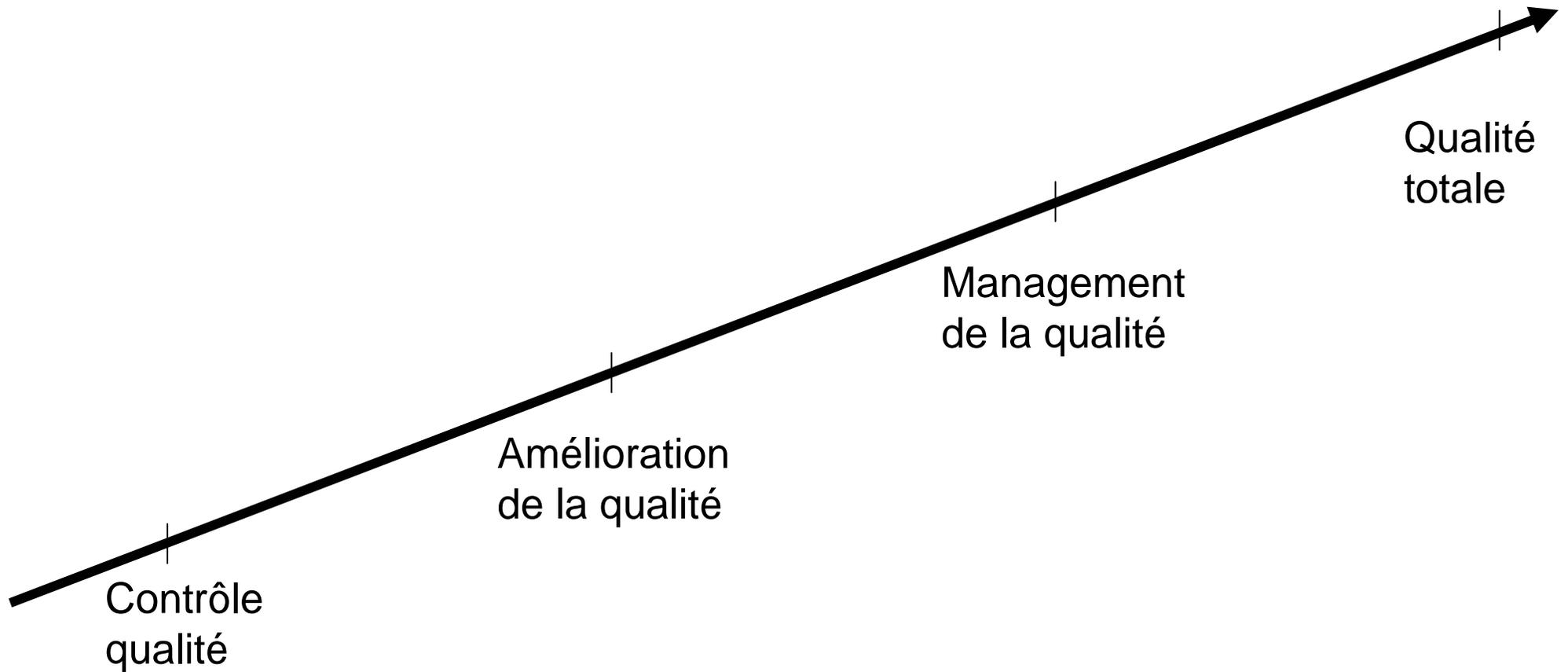


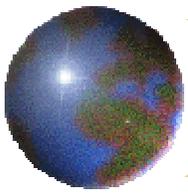
Les acteurs de la qualité

- ❖ **Directeur qualité**
 - ❑ Établit les procédures (plan d'assurance qualité, manuels, etc.) au niveau de l'entreprise
- ❖ **Ingénieur qualité**
 - ❑ Représente le directeur qualité au sein d'une entité
 - ❑ Garant de la bonne application des procédures
- ❖ **Auditeur**
 - ❑ Définit les méthodologies pour l'amélioration de la qualité et du système de management de la qualité
- ❖ **Contrôleur qualité**
 - ❑ Évalue la qualité du produit ou du service et supervise les actions qualité



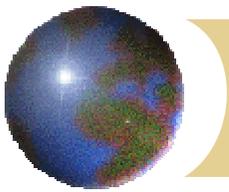
Les approches qualité





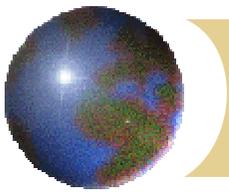
Gestion de la qualité totale

- ✚ TQM (Total Quality Management)
- ✚ Améliorer constamment tous les processus, internes et externes, qui contribuent au produit
- ✚ Utiliser une approche systémique



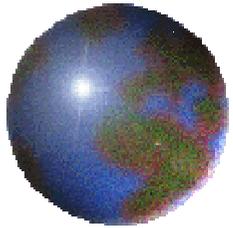
14 principes de la qualité totale [Deming]

- ⊕ Garder en vue l'objectif d'améliorer constamment les produits et les services
- ⊕ Adopter la nouvelle philosophie
- ⊕ Mettre fin à la dépendance à l'égard des inspections et contrôles
 - ⊞ En intégrant la qualité au produit ou au service
- ⊕ Mettre un terme à la pratique des achats au plus bas prix
 - ⊞ Réduire plutôt le coût total en traitant avec un fournisseur unique sur la base d'une relation long terme loyale et confiante
- ⊕ Améliorer constamment le système de production
- ⊕ Etablir un système de formation
- ⊕ Adopter et instituer le leadership et dynamiser l'encadrement
- ⊕ Faire disparaître la peur pour que chacun travaille plus efficacement
- ⊕ Eliminer les barrières entre les services
- ⊕ Eliminer les slogans et objectifs de rendement
 - ⊞ Les problèmes de qualité résident dans le système lui-même, au-delà du pouvoir du personnel
- ⊕ Eliminer les quotas de production
- ⊕ Supprimer les obstacles à la fierté du travail
- ⊕ Encourager l'éducation et l'amélioration de chacun
- ⊕ Agir pour accomplir la transformation

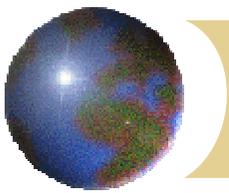


Quelques outils de la qualité

- ✚ Démarches et cadres de qualité
 - ▣ Roue de Deming, Méthode 6 Sigma, Cercle de qualité, démarche 8D, GQM
- ✚ Outils de « défrichage »
 - ▣ Diagramme KJ, brainstorming, feuille de relevé
- ✚ Recherche des causes de problèmes de qualité et des impacts
 - ▣ Ishikawa, Pareto, Histogramme, QQOQCCP
- ✚ Analyse des données et des mesures
 - ▣ Graphique de contrôle, diagramme de dispersion, stratification
- ✚ Analyse du fonctionnement d'un processus/produit
 - ▣ Logigramme, modélisation de processus, graphe PERT
- ✚ Choix d'une solution parmi plusieurs
 - ▣ Matrice de compatibilité, arbre de décision, vote pondéré simple/multiple
- ✚ Optimisation et sécurisation d'un processus
 - ▣ AMDEC, Gantt, Kanban, 5S, Kaizen, Lean

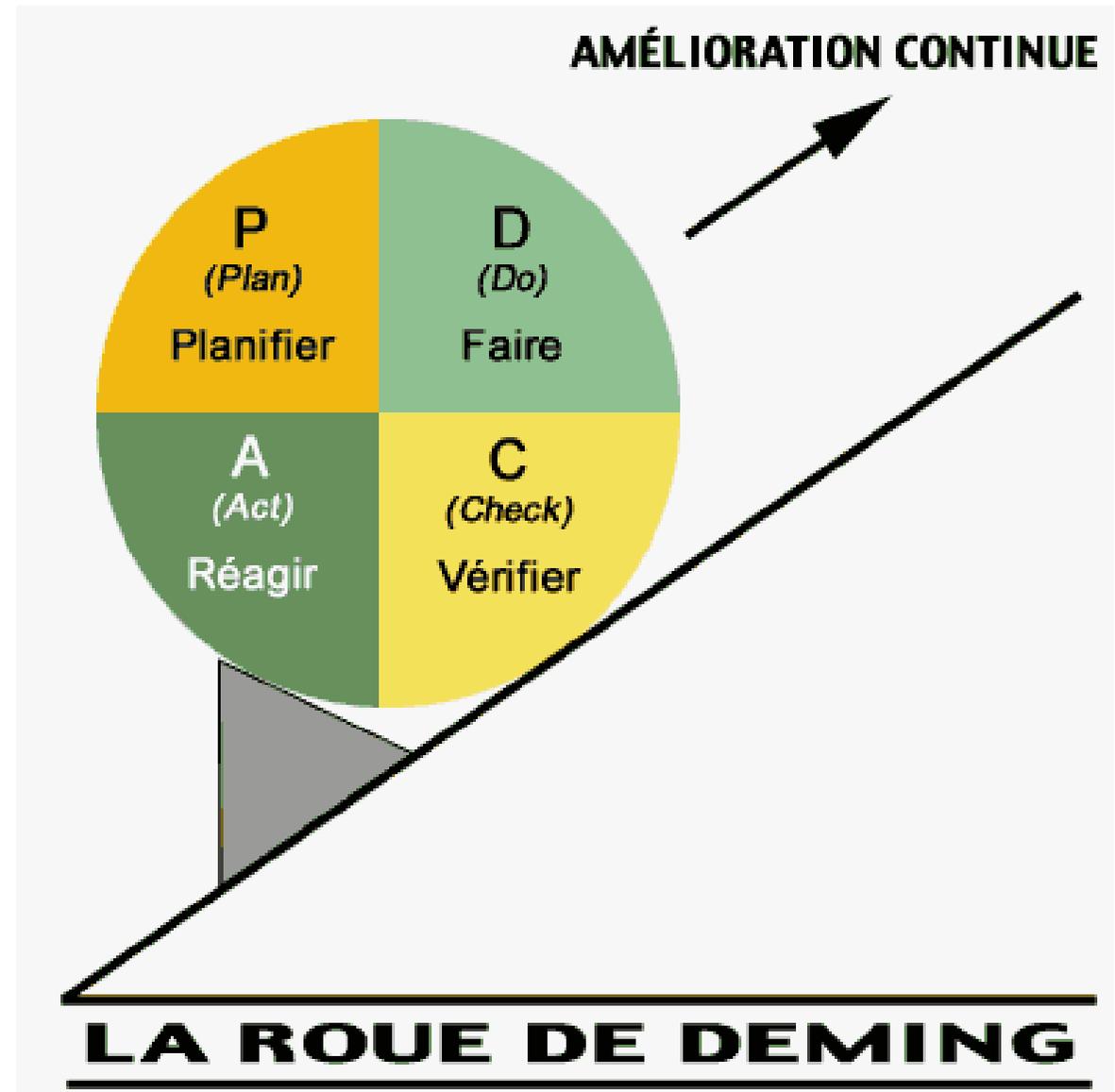


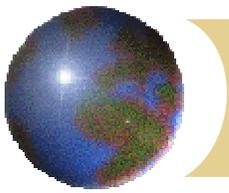
Démarches et cadres de qualité



Plan, Do, Check, Act – roue de Deming

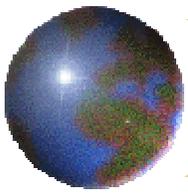
- Méthode séquentielle de conduite de projet permettant d'exécuter un travail de manière efficace et rationnelle
- Comprend 4 étapes P, D, C, A
- S'applique à tous les processus :
 - ▣ Définir un plan d'amélioration
 - ▣ Choisir les méthodes et outils
 - ▣ Mesurer les résultats
 - ▣ Ajuster les actions d'amélioration





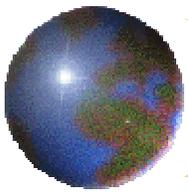
La méthode 6 Sigma (σ)

- ✚ *Origine : Motorola*
- ✚ *Principe : faire en sorte que tous éléments du processus étudié, soient compris dans un intervalle s'éloignant au maximum de 6 fois l'écart type par rapport à la moyenne générale*
- ✚ *Travailler sur le processus afin que seuls des produits conformes aux exigences soient livrés*
- ✚ *Acteurs : pyramide de fonctions d'expertise croissante*
 - *Le Green Belt (« ceinture verte »), consacre 25% de son temps à la conduite de projets d'amélioration*
 - *Le Black Belt (« ceinture noire »), chef d'équipe, se consacre à plein temps à l'amélioration*
 - *Le Master Black Belt, mentor et formateur de Blacks Belts*
 - *Le Deployment Leader ou Champion, chargé d'élaborer la stratégie, le contenu de la formation, les budgets, etc.*
- ✚ *Démarches associées DMAIC (Define, Measure, Analyze, Improve, Control), DMADV (Define, Measure, Analyze, Design, Verify), etc.*



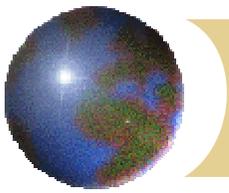
Démarche 8D (ou 8 Do)

- ⊕ D1 : Préparer le process 8D
- ⊕ D2 : Décrire le problème
- ⊕ D3 : Identifier et mettre en place des actions immédiates
- ⊕ D4 : Identifier les vraies causes
- ⊕ D5 : Valider des actions correctives permanentes
- ⊕ D6 : Mettre en œuvre les actions correctives permanentes
- ⊕ D7 : Prévenir toute récurrence
- ⊕ D8 : Féliciter l'équipe



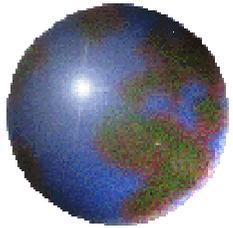
Cercle de qualité

- ❖ Outil de communication
- ❖ Pour partager l'information
- ❖ Au niveau d'une unité ou transversal, temporaire ou permanent, soutien hiérarchique
- ❖ Favoriser la compréhension des objectifs de qualité
- ❖ Permettre la reconnaissance mutuelle
- ❖ 3 facteurs clés de succès
 - ❑ Volonté d'amélioration
 - ❑ Climat de confiance et de transparence
 - ❑ « Légaliser » la critique



Approche par les buts *(Goal-Question-Metric GQM)*

- ❖ Quel est le but de qualité ?
 - ❖ Améliorer la satisfaction du client
- ❖ Quelle dimension de qualité faut-il viser ?
 - ❖ Obtenir des informations fiables sur le client
- ❖ En quels facteurs se décline-t-elle ?
 - ❖ Complétude
 - ❖ Fraîcheur
- ❖ Avec quelles métriques peut-on mesurer ces facteurs ?
- ❖ Quelles sont les valeurs actuelles de qualité ?
- ❖ Comment peut-on améliorer la qualité ?



Outils de « défrichage »

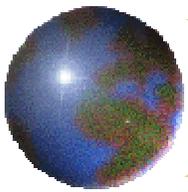
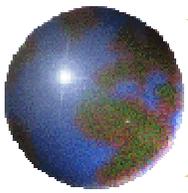


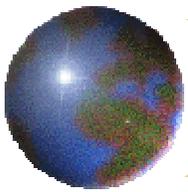
Diagramme d'affinités (ou KJ)

- ✚ De son inventeur : Kawakita Jiro
- ✚ Définir ensemble le problème
- ✚ Chacun écrit ses 3 ou 5 idées ou explications
- ✚ Mise en commun sur un tableau
- ✚ Structuration
- ✚ Hiérarchisation
- ✚ Sélection des catégories à cibler
- ✚ Répartition des tâches



Le « brainstorming »

- ✿ Travail de groupe exprimant le maximum d'idées notées sur un tableau visible de tous
- ✿ 3 phases :
 - ▣ Recherche : expression des idées sans restriction
 - ▣ Regroupement et combinaison d'idées
 - ▣ Conclusion : analyse des causes suspectées



Feuille de relevé

Feuille de relevé

- réponse structurée, préparée sous forme de collecte et d'analyse des données.
- outil générique qui peut être adapté pour une large variété de fins.

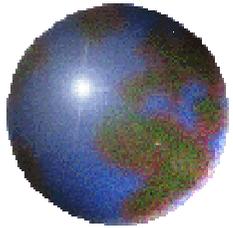
Principe :

- Définir l'événement ou le problème à observer
- Décider du moment où les données seront collectées et pour combien de temps.
- Concevoir le formulaire
- Mettre en place

Exemple : communications téléphoniques

Telephone Interruptions

Reason	Day					Total
	Mon	Tues	Wed	Thurs	Fri	
Wrong number	+++			+++	+++	20
Info request						10
Boss	+++		+++			19
Total	12	6	10	8	13	49



Outils de recherche de causes et d'impacts

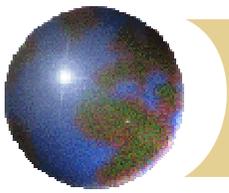
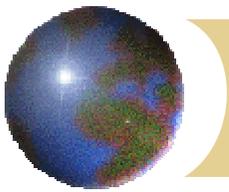


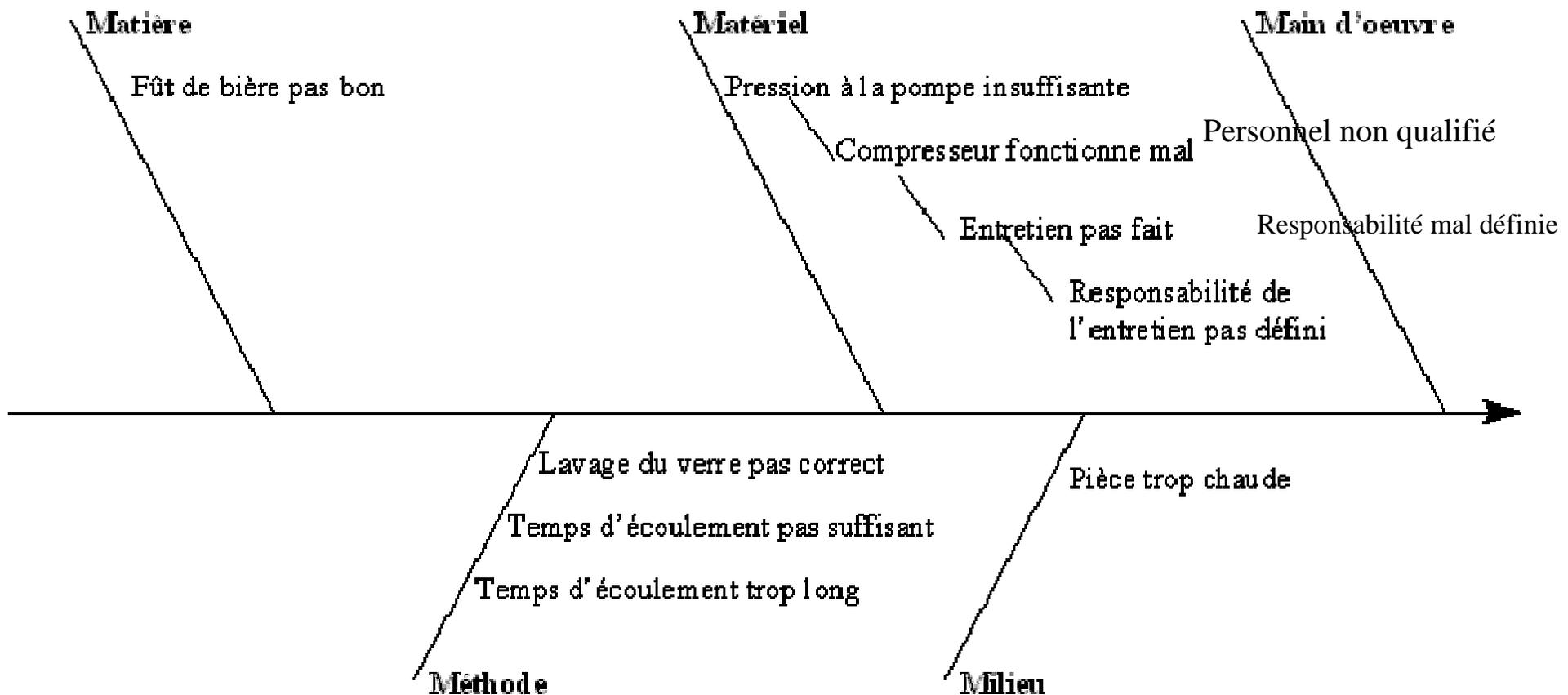
Diagramme d'Ishikawa

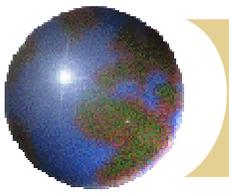
- ❊ Diagramme causes-effets (en arête de poisson)
- ❊ Sert à comprendre les causes d'un défaut de qualité
- ❊ Analyse le rapport existant entre un problème et ses causes
- ❊ Fondé sur un travail de groupe
- ❊ Trouver toutes les causes possibles au défaut de qualité et les classer en 5 familles (5M):
 - ❑ Matières, Milieu, Méthodes, Matériels, Main d'oeuvre



Exemple de diagramme d'Ishikawa

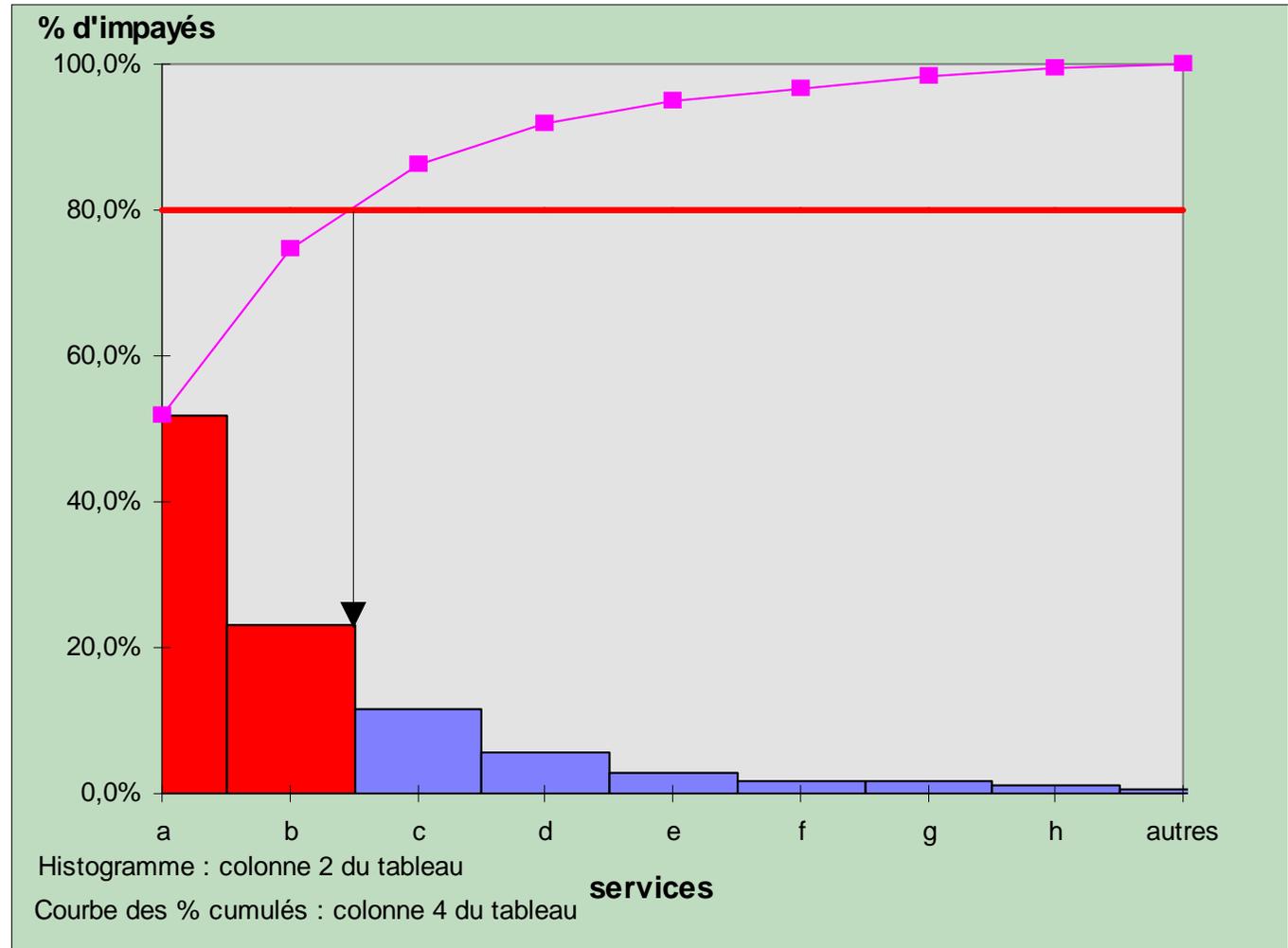
- La bière que je vends n'est pas bonne

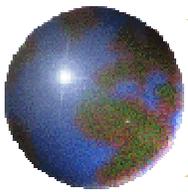




Graphe de Pareto

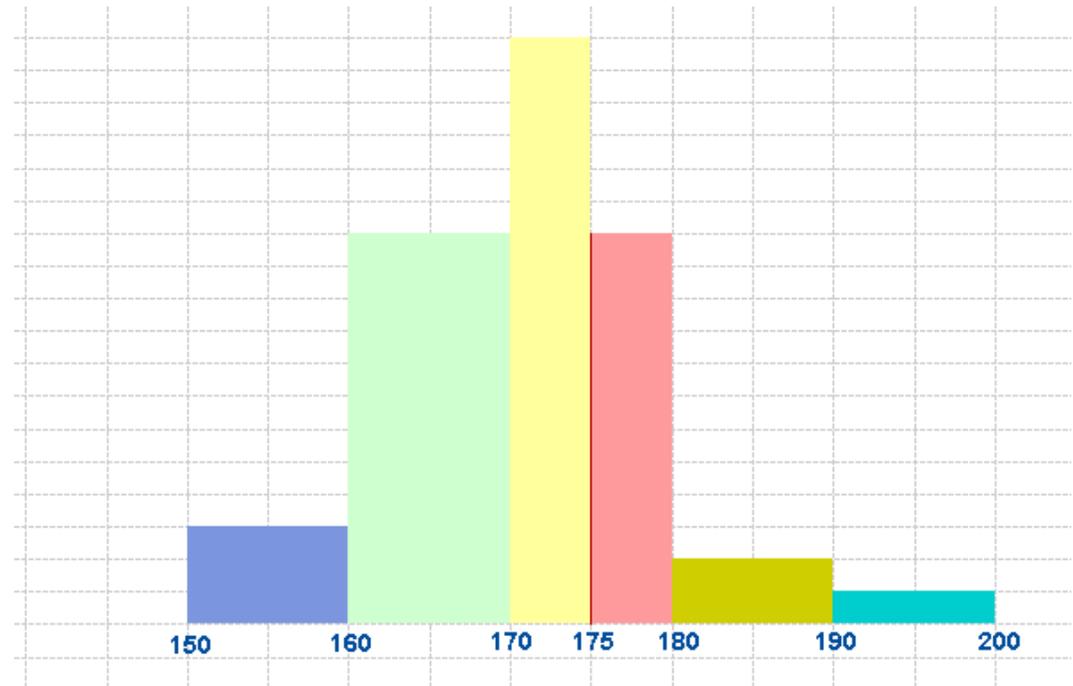
- Permet de classer les phénomènes par ordre d'importance
- Histogramme décroissant avec une ligne de cumul

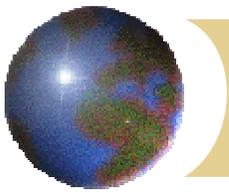




Histogramme

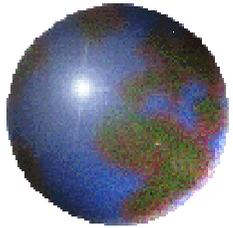
- ❖ Permet d'étudier une distribution de données
- ❖ Et de la comparer à une distribution standard
- ❖ Il existe une typologie d'histogramme (normal, bimodal, plateau, etc.)



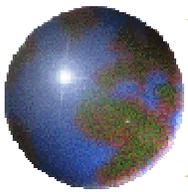


Le QQOQCCP

- ❖ Why, what, where, when, who, how, how much ?
- ❖ Technique de recherche d'informations sur un problème et ses causes en se posant les questions: quoi, qui, où, quand, comment, pourquoi
- ❖ Pour chaque question se demander combien ?

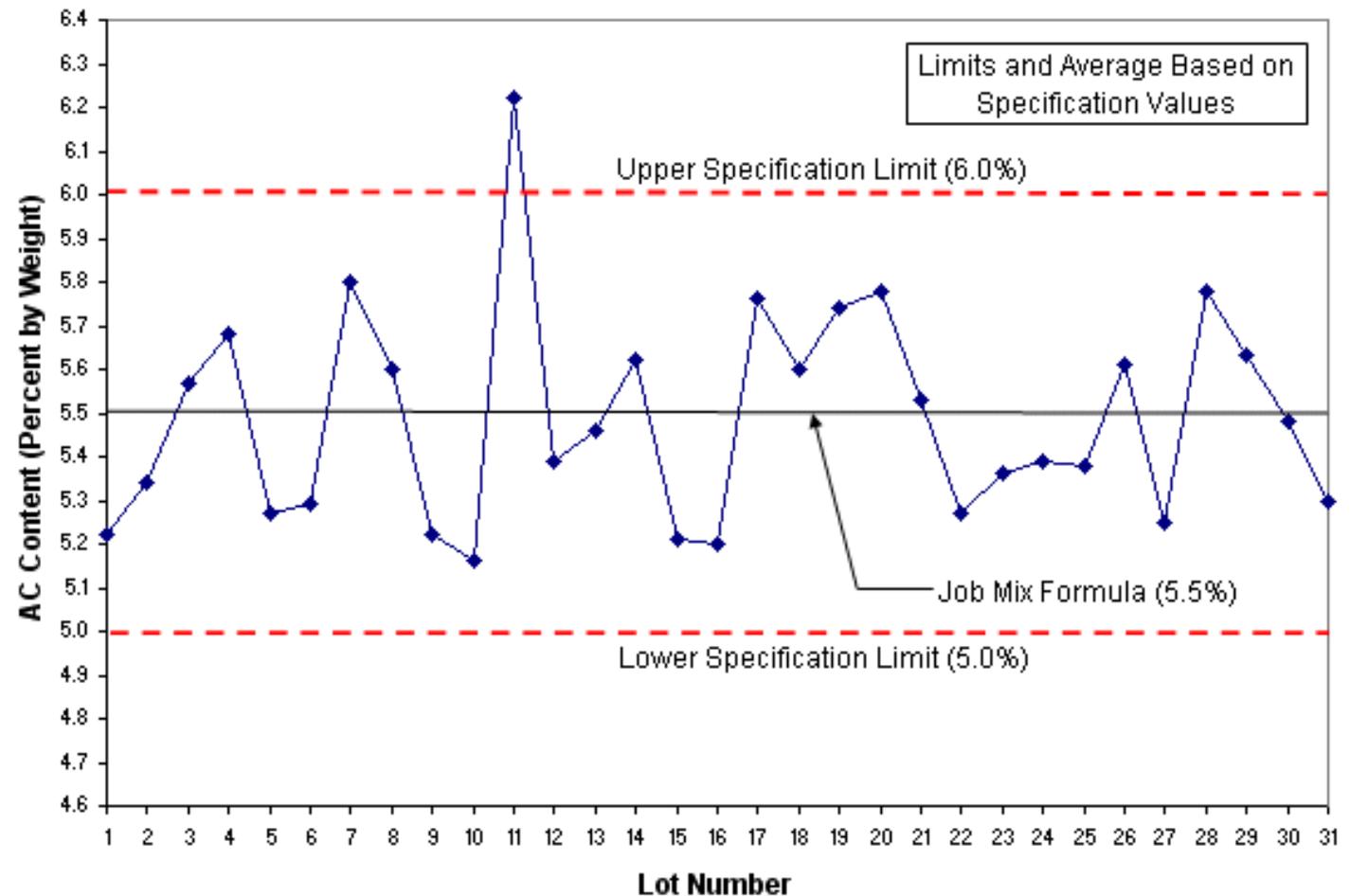


Outils d'analyse des données et des mesures



Graphique de contrôle

- Utilisé pour étudier un processus dans le temps
- Les données sont tracées par ordre chronologique.
- ligne centrale = moyenne
- ligne supérieure et ligne inférieure déterminées à partir des données historiques.



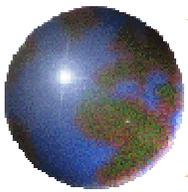
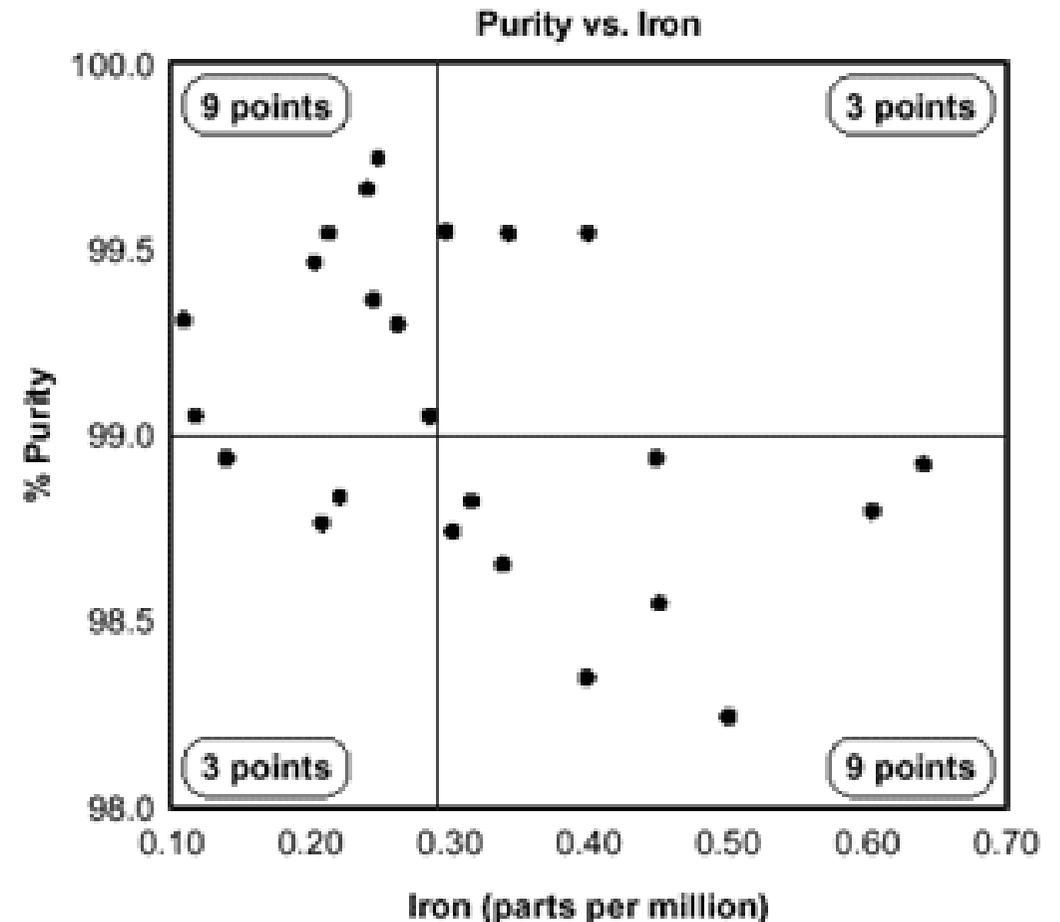
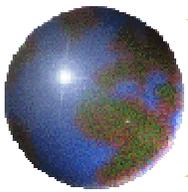


Diagramme de dispersion

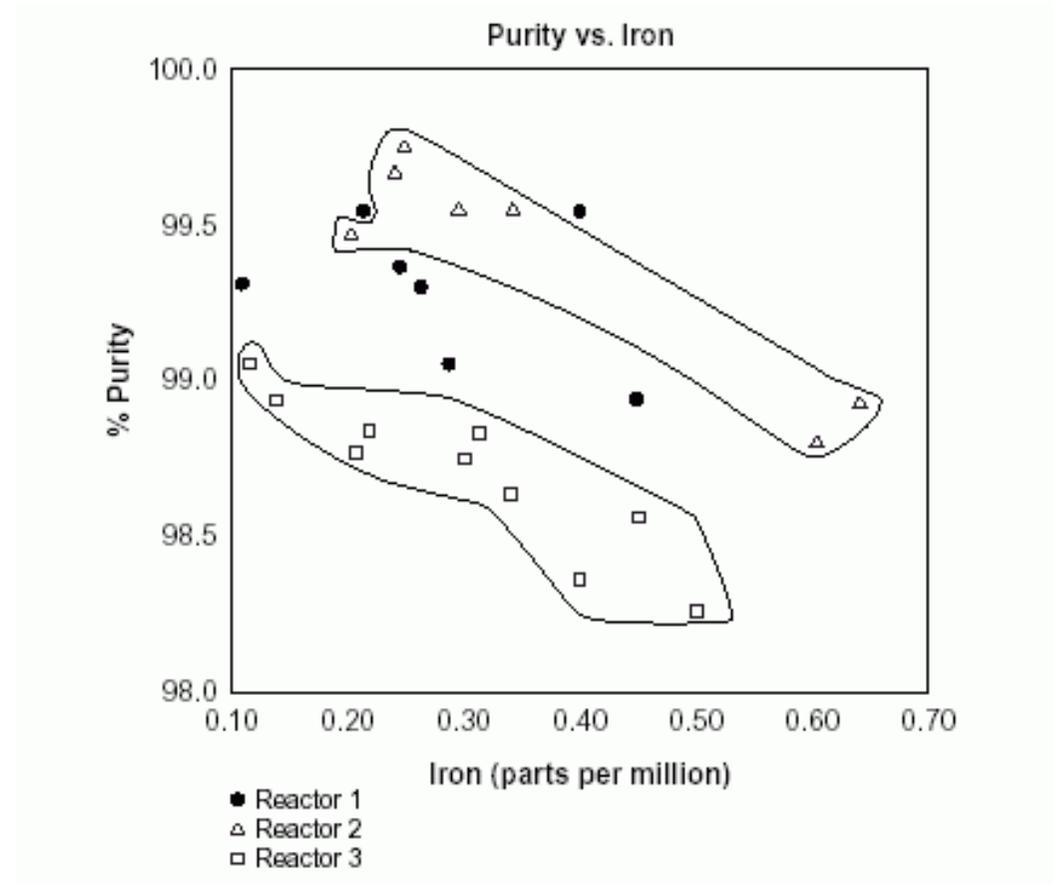
- ➊ Moyen simple de vérifier une relation entre deux variables
- ➋ Plus le nuage de points se rapproche d'une droite, plus la relation est forte
- ➌ Corrélation n'est pas causalité

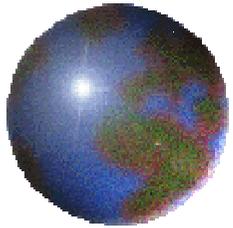




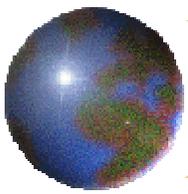
La stratification

- Permet de distinguer les données provenant de différentes sources pour analyser des « patterns »
- Aussi appelé « flow chart » ou « run chart »



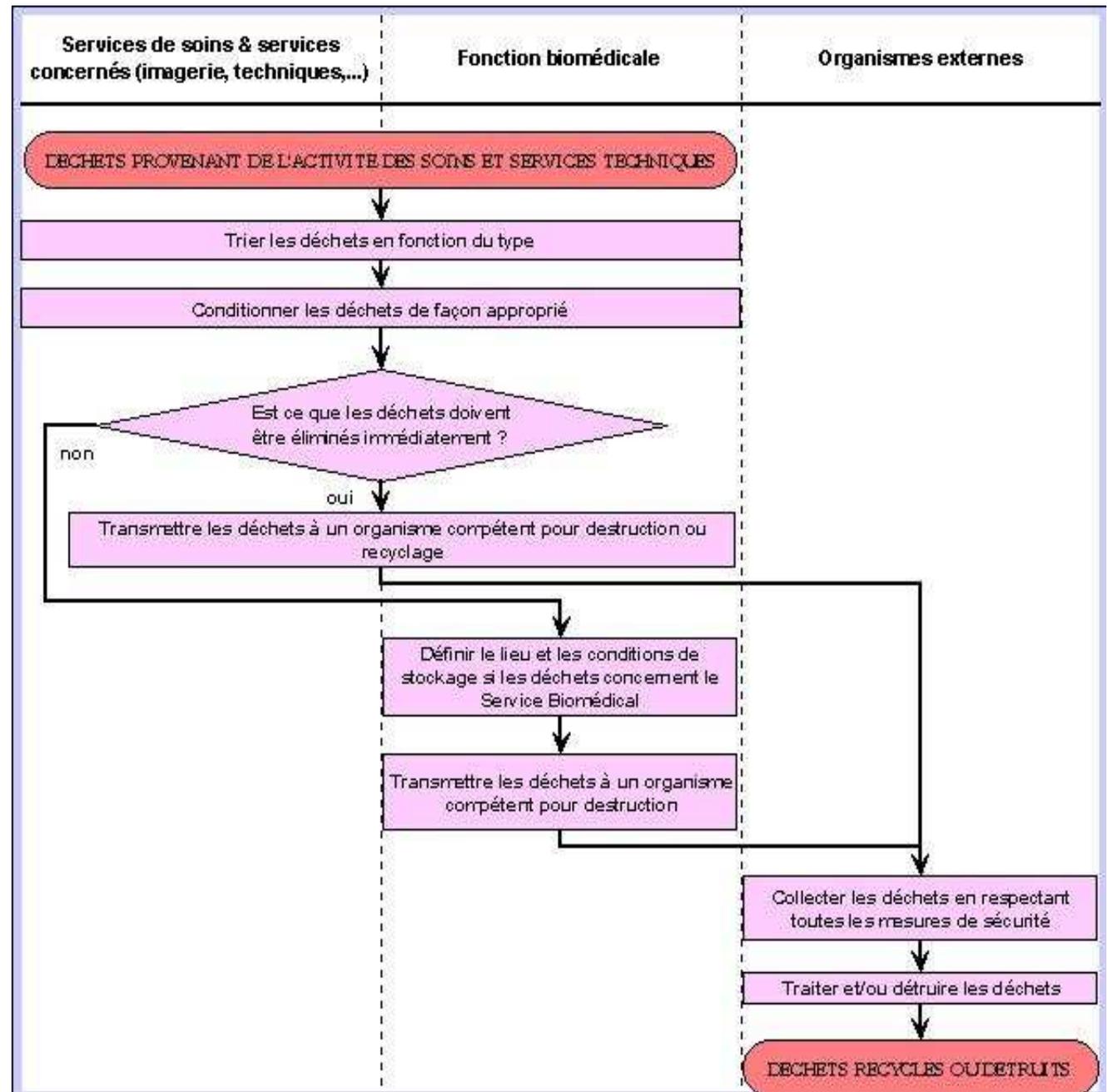


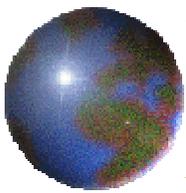
*Outils d'analyse
du fonctionnement d'un
processus/produit*



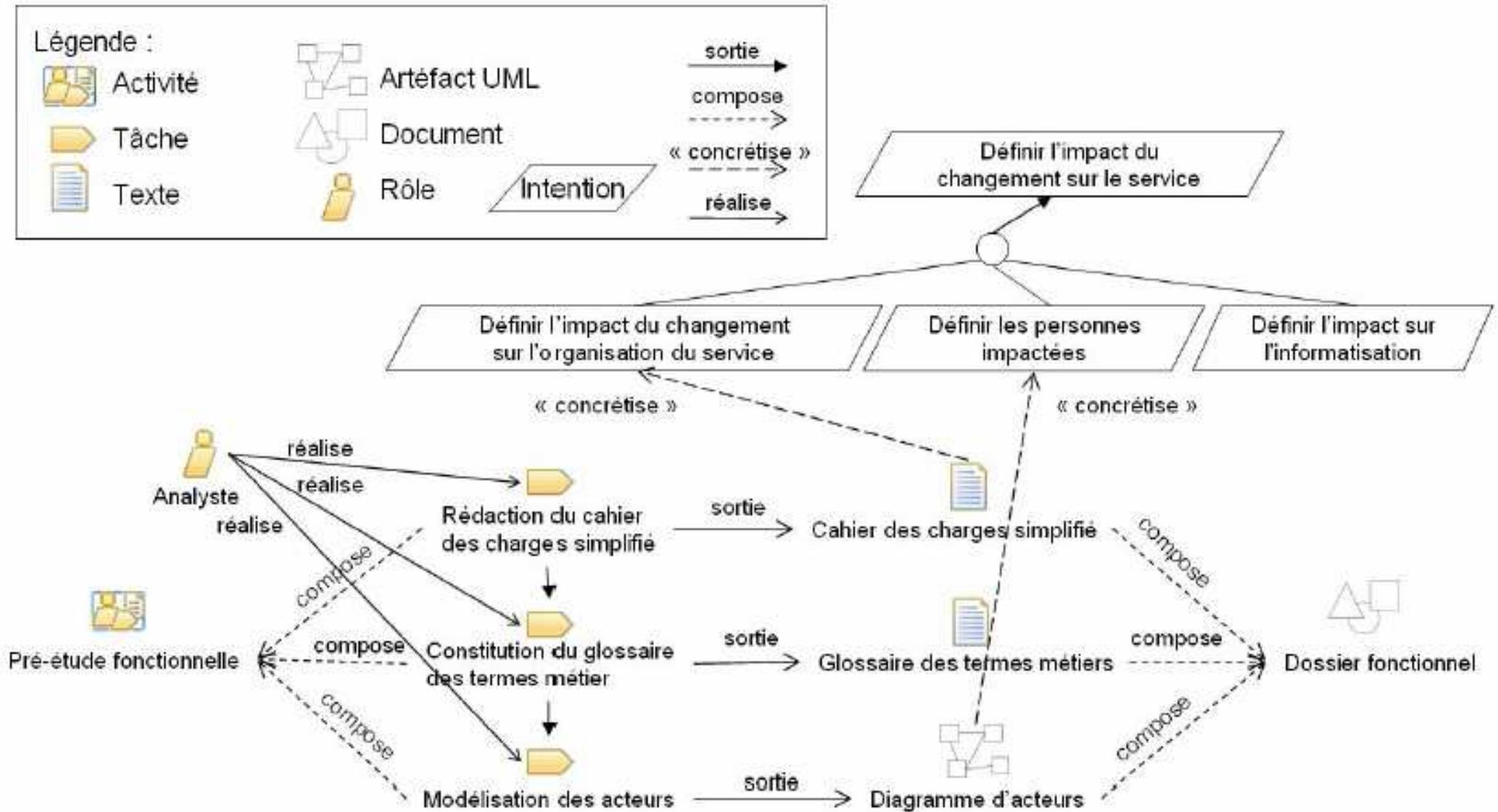
Le logigramme

- Représentation de la procédure qualité

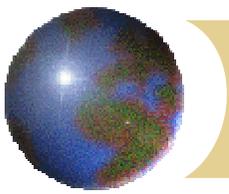




Modélisation de processus métier



Modèle de processus avec les formalismes KAOS et SPEM.

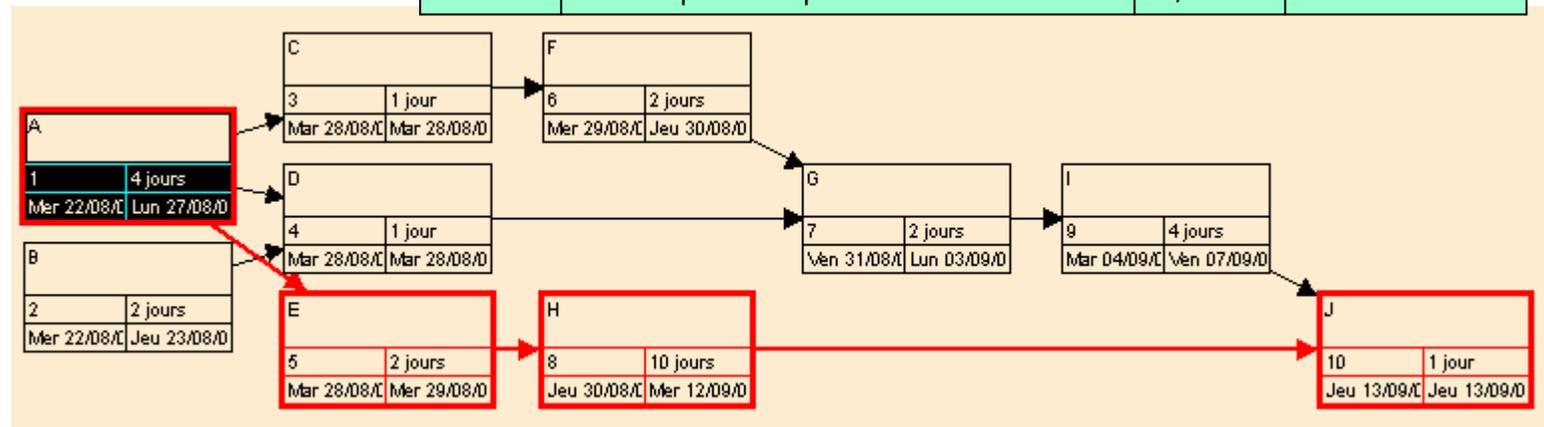


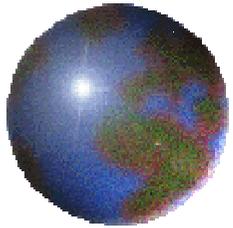
Outil PERT

(Project Evaluation and Review Technique)

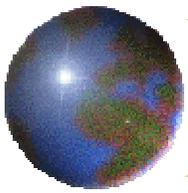
- Représentation des étapes d'un processus et des contraintes entre étapes
- Exemple : La construction d'un entrepôt est découpée en dix tâches dont les caractéristiques sont données dans le tableau ci-contre
- Graphe PERT**

tâches	nature	prédecesseurs	durée en jours
A	acceptation des plans par le propriétaire		4
B	préparation du terrain		2
C	commande des matériaux	A	1
D	creusage des fondations	A, B	1
E	commande des portes et fenêtres	A	2
F	livraison des matériaux	C	2
G	coulage des fondations	D, F	2
H	livraison des portes et fenêtres	E	10
I	pose des murs, de la charpente et du toit	G	4
J	mise en place des portes et fenêtres	H, I	1





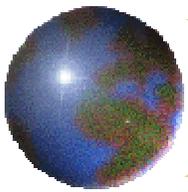
Outils d'aide au choix



La matrice de compatibilité

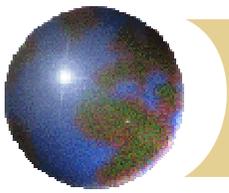
- ⊕ Outil d'aide à la décision
- ⊕ Permet de faire un choix parmi plusieurs propositions
- ⊕ Croisement des critères de choix (coûts, délais, efficacité,...) et des sujets (problèmes à sélectionner ou solutions à retenir)
- ⊕ Les sujets sont comparés entre eux en fonction de leurs scores
- ⊕ Les problèmes ou solutions ne répondant pas à un ou plusieurs critères sont éliminés
- ⊕ Ceux répondant à la plupart ou à tous sont retenus
- ⊕ Exemple organisation d'un déplacement professionnel

	Train	Avion	Bateau	Car
Coût (moins de 5000€)	+	-	+	+
Rapidité (moins de 4h)	+	+	-	-
Confort	+	+	?	-
Sécurité	+	+	+	-



Le vote pondéré

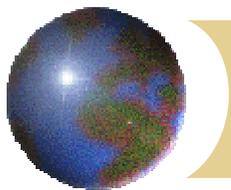
- ✚ Outil de choix lorsque les données sont qualitatives
- ✚ Sélection finale du problème que le groupe souhaite résoudre en premier à partir d'un résultat d'un vote simple
 - ▣ Vote pondéré multicritère
 - ▣ Vote pondéré simple



Exemple de vote pondéré simple

✦ Comment diminuer les frais de chauffage ?

	Pierre	Paul	Jack	TOTAL
Mettre des fenêtres et portes isolantes	3	4	3	10
Mettre de portes a fermeture automatique	4	2	1	7
Chauffer moins cette année	1	1	2	4
Changer les radiateurs	2	3	4	9

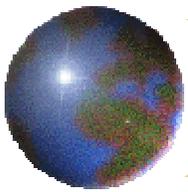


Exemple de vote pondéré multiple

Comment diminuer les frais de chauffage ?

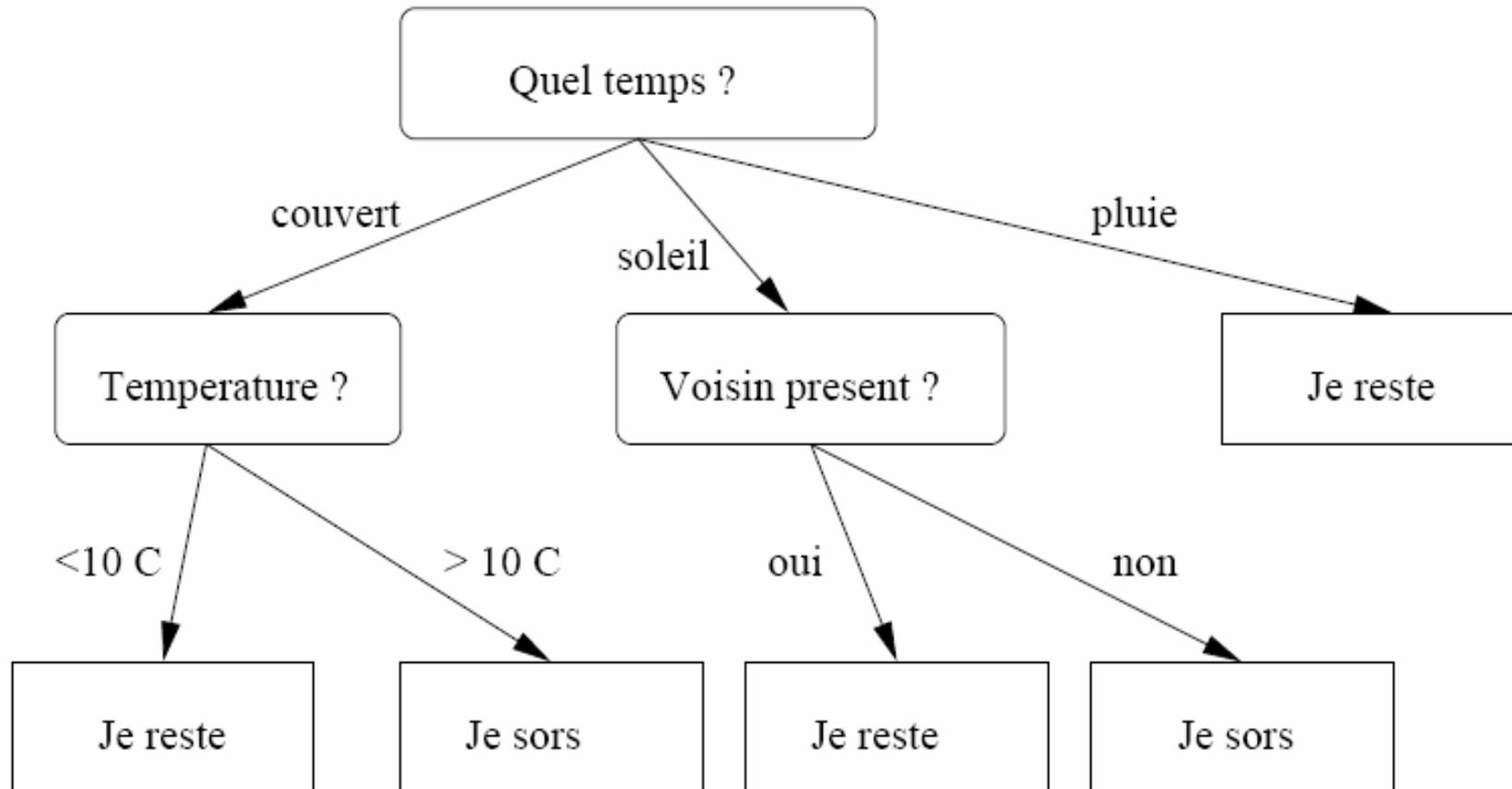
	Mettre des fenêtres et portes isolantes	Mettre de portes a fermeture automatique	Chauffer moins cette année	Changer les radiateurs
Coûts $\rightarrow \times 5$	$2+2+1 \Rightarrow 25$	$3+4+4 \Rightarrow 55$	$4+3+3 \Rightarrow 50$	$1+1+2 \Rightarrow 20$
Efficacité $\rightarrow \times 4$	$3+4+4 \Rightarrow 44$	$2+2+1 \Rightarrow 20$	$1+1+2 \Rightarrow 16$	$4+3+3 \Rightarrow 40$
Délai de mise en place $\rightarrow \times 3$	$3+4+3 \Rightarrow 30$	$2+2+2 \Rightarrow 18$	$1+1+1 \Rightarrow 9$	$4+3+4 \Rightarrow 33$
TOTAL	99	93	75	93

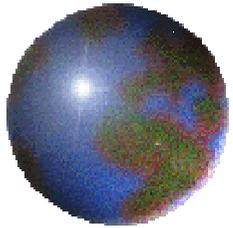
Trois votants : Pierre, Paul et Jack



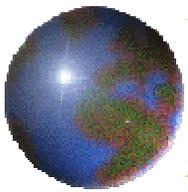
Arbre de décision

Vais-je sortir mon chien ?



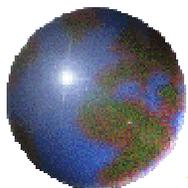


Outils d'optimisation et de sécurisation de processus

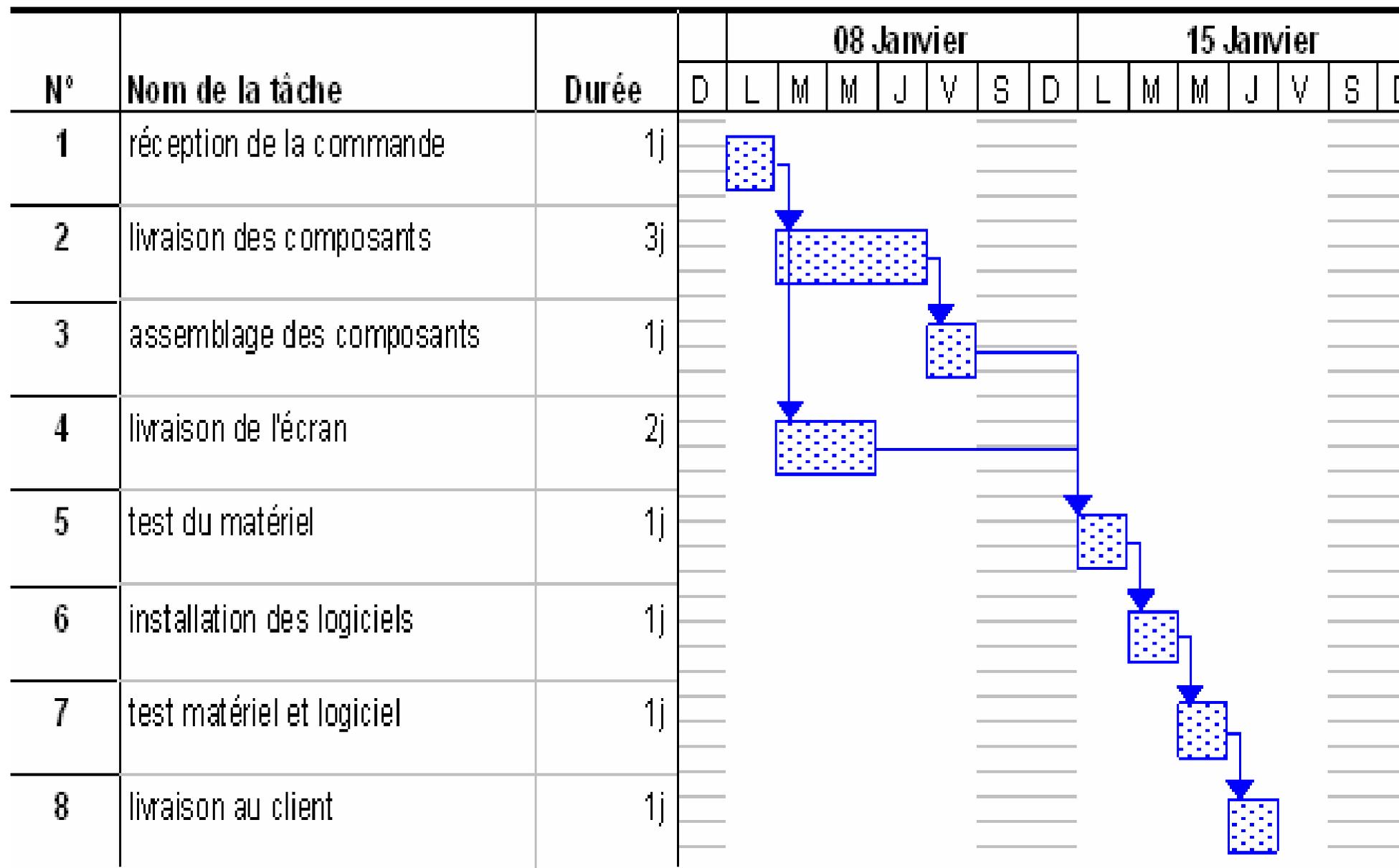


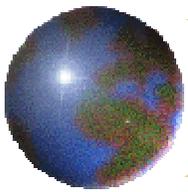
L'AMDEC

- ❖ Analyse des Modes de Défaillances, de leurs Effets et de leur Criticité
- ❖ AMDEC produit, AMDEC processus, AMDEC moyen
- ❖ Recenser toutes les causes potentielles d'une défaillance et évaluer la criticité selon trois notes : G (gravité), O (occurrence), D (détection)
- ❖ G : gravité ou sévérité de l'impact du défaut
- ❖ O : fréquence d'apparition de la cause
- ❖ D : probabilité de non détection de la cause
- ❖ Indice de criticité : $G * O * D$
- ❖ Mettre en place des actions en jouant sur un ou plusieurs des paramètres



Le diagramme de Gantt

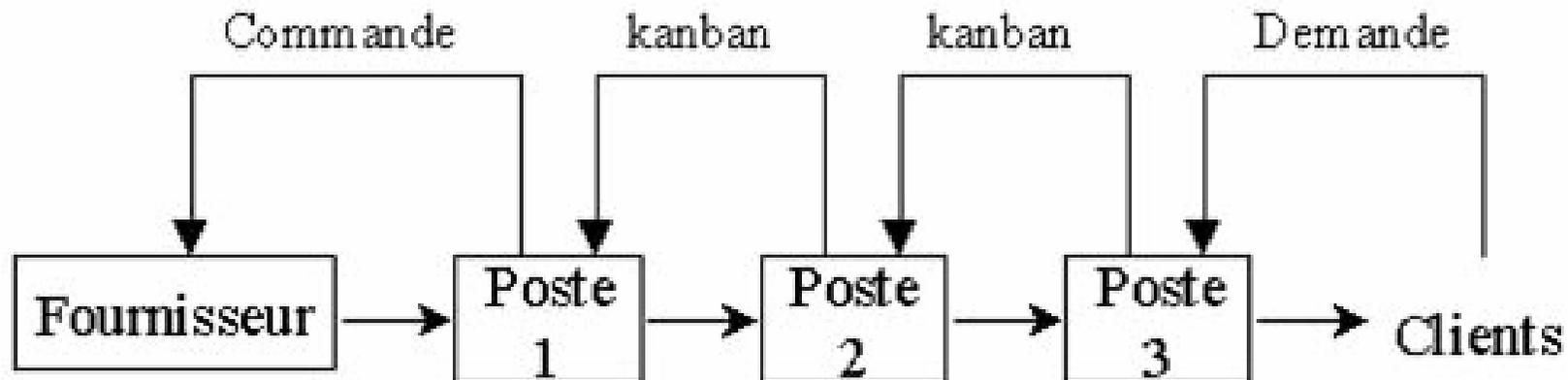


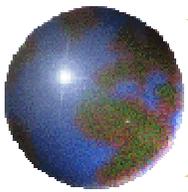


Méthode Kanban

- Méthode de suivi de production utilisée dans le juste-à-temps
- Se fonde sur un système d'étiquettes permettant le juste-à-temps

KANBAN - FLUX TIRE

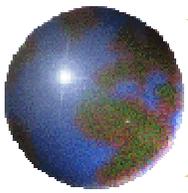




5S

- Outil d'amélioration continue, qui optimise l'organisation d'un poste de travail, d'un service ou d'une entreprise

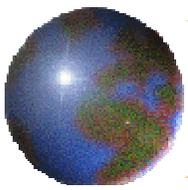
Mot Japonais	Traduction	Interprétation
Seiri	Débarras	Trier
Seiton	Ranger	Classer
Seiso	Nettoyage	Nettoyer
Seiketsu	Ordre	Conservé en ordre et propre
Shitsuke	Rigueur	Formaliser et impliquer



Kaizen : changer pour faire mieux

- ✚ 1 Abandonner les idées fixes, refuser l'état actuel des choses.
- ✚ 2 Au lieu d'expliquer ce que l'on ne peut pas faire, réfléchir comment faire.
- ✚ 3 Réaliser aussitôt les bonnes propositions d'amélioration.
- ✚ 4 Ne pas chercher la perfection, gagner 60% maintenant.
- ✚ 5 Corriger l'erreur immédiatement sur place.
- ✚ 6 Trouver des idées dans la difficulté.
- ✚ 7 Chercher la cause réelle, respecter les « 5 pourquoi ? » et chercher ensuite la solution.
- ✚ 8 Prendre en compte les idées de 10 personnes au lieu d'attendre l'idée géniale d'une seule.
- ✚ 9 Essayer et ensuite valider.
- ✚ 10 L'amélioration est infinie.





Lean

➤ Méthode Toyota

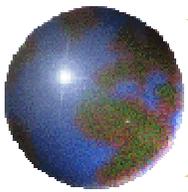
➤ 14 principes :

- ❑ Penser sur le long terme
- ❑ Fluidité
- ❑ Flux tirés
- ❑ Production constante et lissée
- ❑ Automatisation avec une touche humaine
- ❑ Tâches standardisées
- ❑ Contrôles visuels
- ❑ Technologies et méthodes fiables
- ❑ Cultiver les leaders
- ❑ Faire monter en compétences les personnes de qualité
- ❑ Respecter et motiver ses partenaires
- ❑ Aller toujours sur le terrain
- ❑ Prendre les décisions en consensus
- ❑ Amélioration continue

➤ Application en systèmes d'information : méthodes agiles

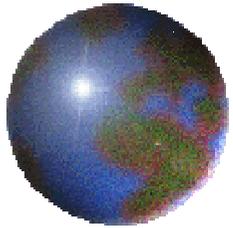
And what do you think about lean?



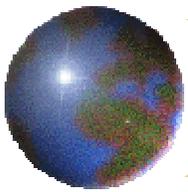


Les noms de la qualité

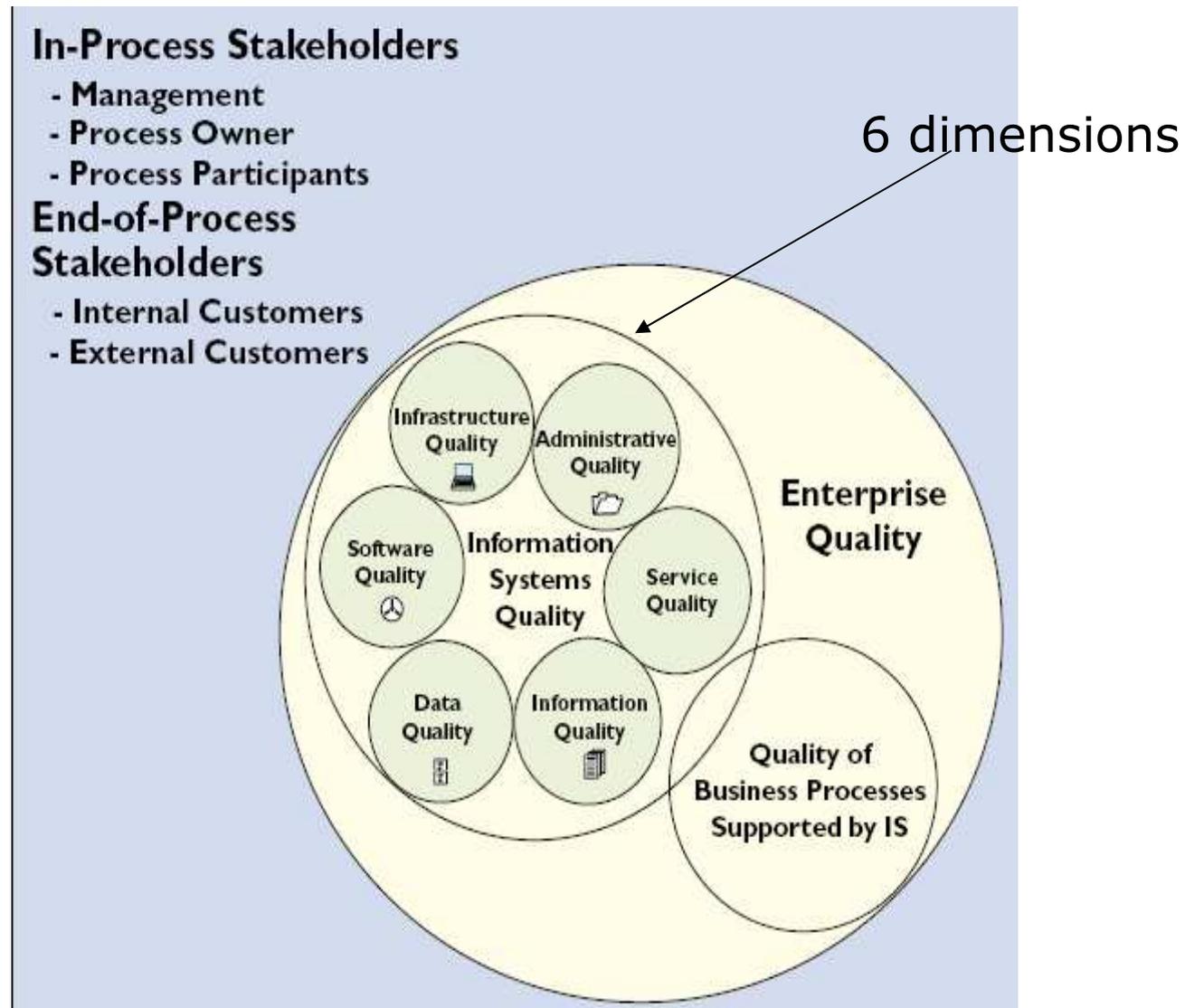
- ❖ Deming : roue PDCA, qualité totale, insiste sur la cohérence de l'approche «assurer la qualité, c'est donner aux clients des garanties sur sa capacité à fournir un produit ou un service conformément à ses attentes et dans la durée »
- ❖ Juran : a diffusé le principe de Pareto ou loi des 80/20, a initialisé des formations, la qualité est « la meilleure adéquation au besoin »
- ❖ Crosby : la qualité, « conformité à des exigences », conduit à la chute des coûts et à l'accroissement de la productivité
- ❖ Taguchi : propose des plans d'expérience et des tables pour évaluer et améliorer la qualité des produits

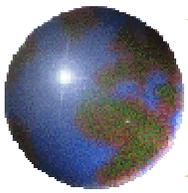


3. Qualité des systèmes d'information



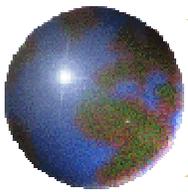
Qualité des systèmes d'information





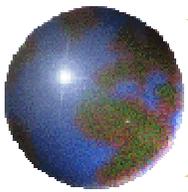
Dimensions de qualité (1/2)

- ❖ Qualité de l'infrastructure
 - ❑ Matériel
 - ❑ Logiciel de base
 - ❑ Réseau
 - ❑ Système d'exploitation, etc.
- ❖ Qualité du logiciel
 - ❑ Qualité des applications construites et/ou maintenues
- ❖ Qualité des données
 - ❑ Qualité des données entrant dans les différents systèmes d'information



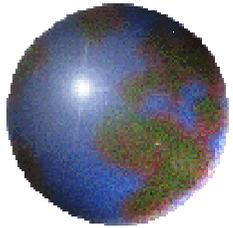
Dimensions de qualité (2/2)

- ❖ Qualité de l'information
 - ❑ Qualité des données sortant des systèmes d'information
- ❖ Qualité administrative
 - ❑ Qualité de la fonction système d'information
 - ❑ Inclut la qualité des processus de :
 - Élaboration du budget
 - Élaboration du planning
- ❖ Qualité de service
 - ❑ Qualité du service rendu vu par le « client »

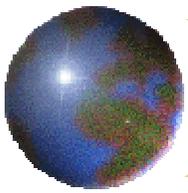


Interactions entre les dimensions

- ✚ Qualité de l'infrastructure -> qualité des données et qualité de service
- ✚ Qualité de l'information sortant d'un SI -> qualité des données entrant dans un SI
- ✚ Etc.

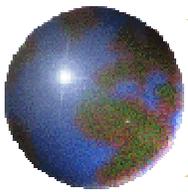


4. Qualité et normalisation



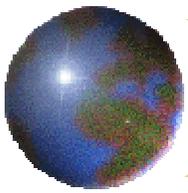
Pourquoi des normes ?

- ❖ Formalisent le fonctionnement et les responsabilités des intervenants et par là :
 - ❑ Renforcent l'implication du management
 - ❑ Gèrent l'après-certification
 - ❑ Motivent les acteurs
 - ❑ Satisfont le client
 - ❑ Donnent la priorité à l'efficacité et non à la conformité des procédures
 - ❑ Répondent à la certification



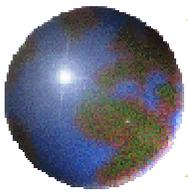
Les normes ISO 9000

- Créées en 1987
- 2 révisions en 1994 et 2000
- « outils de progrès »
- 4 normes principales
 - ISO 9000:2000
 - principes essentiels et vocabulaire
 - ISO9001:2000 : QUI et QUOI
 - exigences
 - utilisée pour faire des certifications
 - ISO9004:2000 : COMMENT
 - Lignes directrices pour l'amélioration des performances
 - ISO19011
 - Lignes directrices pour les audits de systèmes de management qualité
- ISO9001 :
 - Qualité externe
 - À usage contractuel
 - Pour la certification des procédés auprès des clients
 - Réalisée par un tiers indépendant rattaché à un organisme de certification



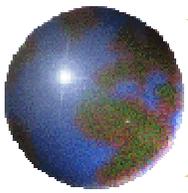
Les 8 principes de management de ISO 9000:2000

- Ecoute client
- Leadership
- Implication du personnel
- Approche processus
- Management par approche système
- Amélioration continue
- Approche factuelle pour la prise de décision
- Relations mutuellement bénéfiques avec les fournisseurs



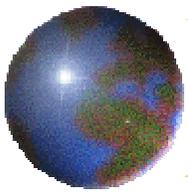
Ecoute client

- ❖ Comprendre les besoins présents et futurs
- ❖ Satisfaire les exigences
- ❖ S'efforcer d'aller au devant des attentes



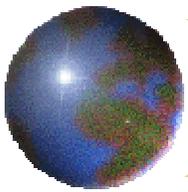
Leadership

- ❖ Les dirigeants doivent:
 - ❑ Établir la finalité et les orientations de l'organisme
 - ❑ Créer les conditions d'implication des personnes dans la réalisation des objectifs de l'organisme



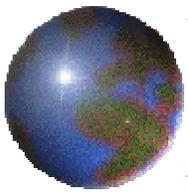
Implication du personnel

- ❖ Motiver le personnel
- ❖ Les membres sont responsables de leur performance
- ❖ Le personnel participe et contribue à l'amélioration continue
- ❖ Le personnel utilise ses aptitudes au profit de l'organisme



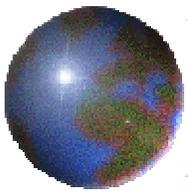
Approche processus

- ⊕ Les ressources et les activités sont gérées comme un processus
- ⊕ Avec une focalisation sur les opportunités d'amélioration
- ⊕ Identifier et gérer les processus, leurs combinaisons et leurs interactions
- ⊕ Points clés :
 - ⊞ La direction est responsable du management de la qualité
 - ⊞ Les ressources : humaines, temps, finances, infrastructures, information, environnement de travail
 - ⊞ Processus de satisfaction du client
 - ⊞ Mesures, analyses et amélioration



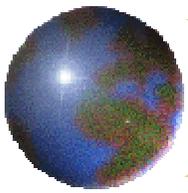
Management par approche système

- ➊ Identifier, comprendre et gérer des processus comme un système
- ➋ Intégration et alignement des processus



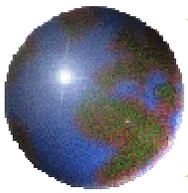
Amélioration continue

- ✚ L'amélioration continue de la performance globale d'un organisme est un objectif permanent
- ✚ Alignement des activités d'amélioration à tous les niveaux
- ✚ Souplesse et rapidité de réaction face aux opportunités



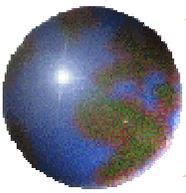
Approche factuelle pour la prise de décision

- ❖ Les décisions efficaces sont fondées sur l'analyse des données et des informations
- ❖ Les décisions bien informées si les données sont exactes et fiables
- ❖ La prise de décision est fondée sur des données factuelles.



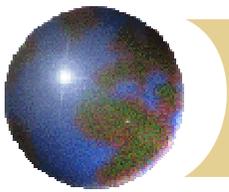
Relations mutuellement bénéfiques avec les fournisseurs

- ❖ Interdépendance de l'organisme et de ses fournisseurs
- ❖ Capacité pour les deux organismes à créer de la valeur
- ❖ Mise en commun des acquis et des ressources avec les partenaires



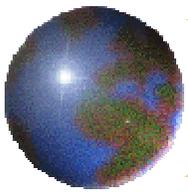
Standards développés ou en cours

- ISO/TS 8000-100:2009 Data quality -- Part 100: Master data: Overview
- ISO/TS 8000-110:2009 Data quality -- Part 110: Master data: Exchange of characteristic data: Syntax, semantic encoding, and conformance to data specification.
- ISO/TS 8000-120:2009 Data quality -- Part 120: Master data: Exchange of characteristic data: Provenance
- ISO/PAS 26183-2006 SASIG Product data quality guidelines for the global automotive industry
- ISO/TR 21707-2008 Intelligent transport systems - Integrated transport information, management and control - Data quality in ITS systems
- ISO/IEC 25012:2008 Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Data quality model
- BS 7986-2005 Data quality metrics for industrial measurement and control systems – Specification
- ISO 19113:2002 Geographic information — Quality principles
- ISO 19114:2003 Geographic information — Quality evaluation procedures
- ISO TS 19138-2006 Geographic information — Data quality measures
- ISO/NP 19157 Geographic information -- Data quality
- ISO/NP TS 19158 Geographic information - Quality assurance of data supply
- International Monetary Fund, IMF: Data Quality Assessment Framework, DQAF



Normes de qualité

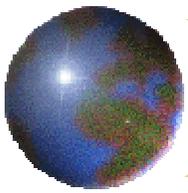




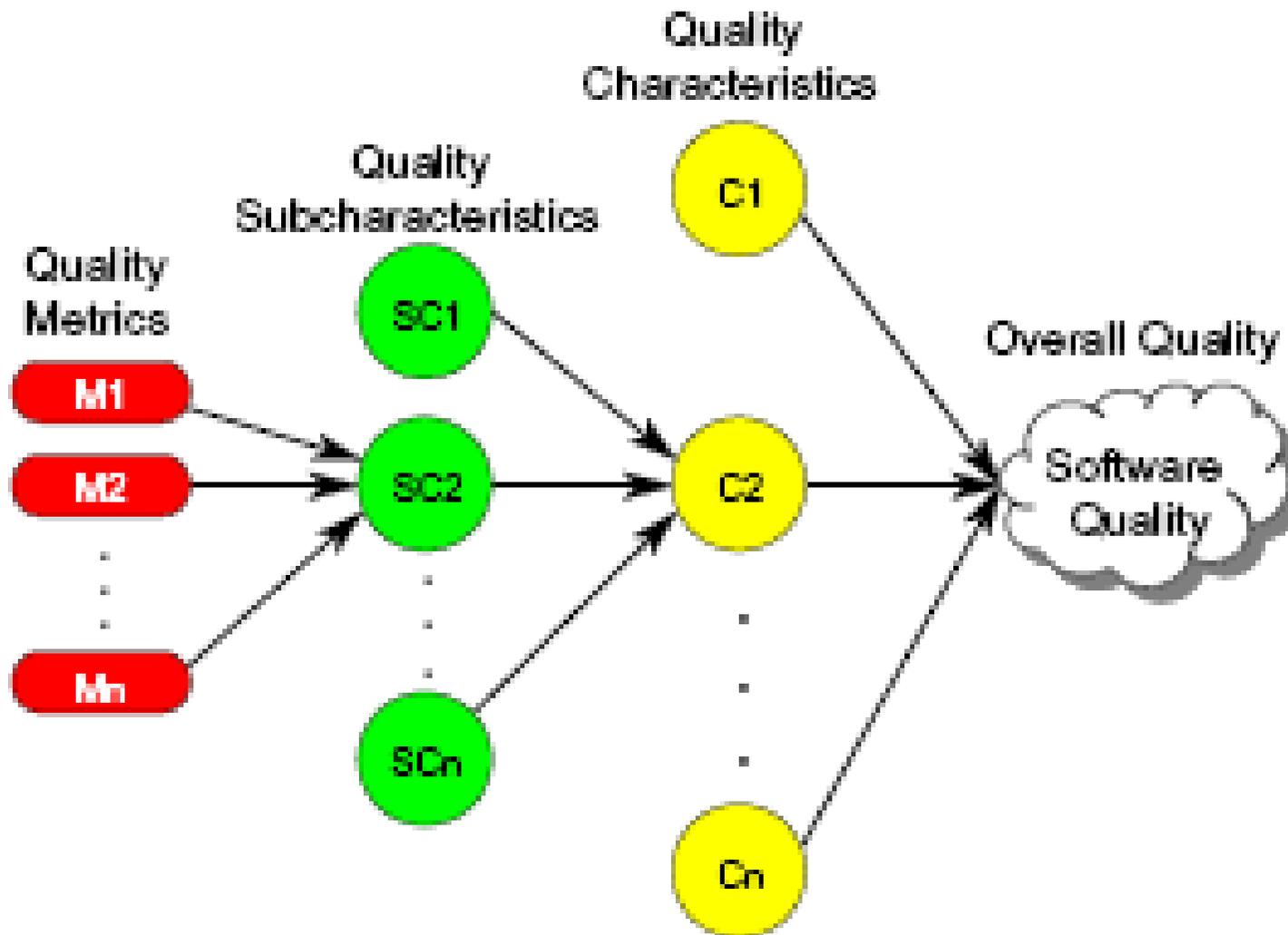
Norme ISO/IEC 9126



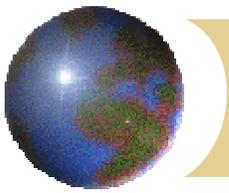
- Process quality: qualité des processus du cycle de vie du logiciel
- Internal quality: qualité des produits intermédiaires, incluant les modèles statiques et dynamiques, la documentation et le code source
- External quality: qualité du système final tel que vérifiable à son comportement externe
- Quality in use: effet du système lors de son utilisation – satisfaction des attentes des utilisateurs



Le modèle de qualité logicielle ISO/IEC 9126



J. Акoкa / J. warrau
6 caractéristiques / 24 sous-caractéristiques / 113 métriques

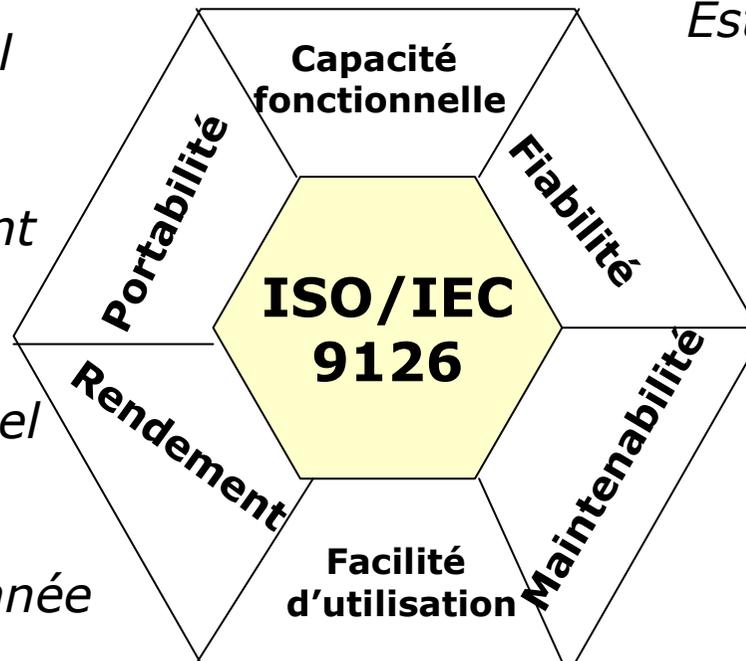


Les 6 caractéristiques de la qualité logicielle ISO/IEC 9126

*Est-ce que le logiciel répond
aux besoins fonctionnels exprimés ?*

*Est-ce que le logiciel
peut être transféré
d'une plate-forme
ou d'un environnement
à un autre ?*

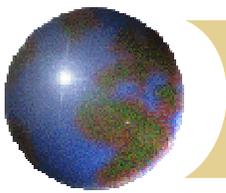
*Est-ce que le logiciel maintient
son niveau de service
dans des conditions
précises et pendant
une période déterminée ?*



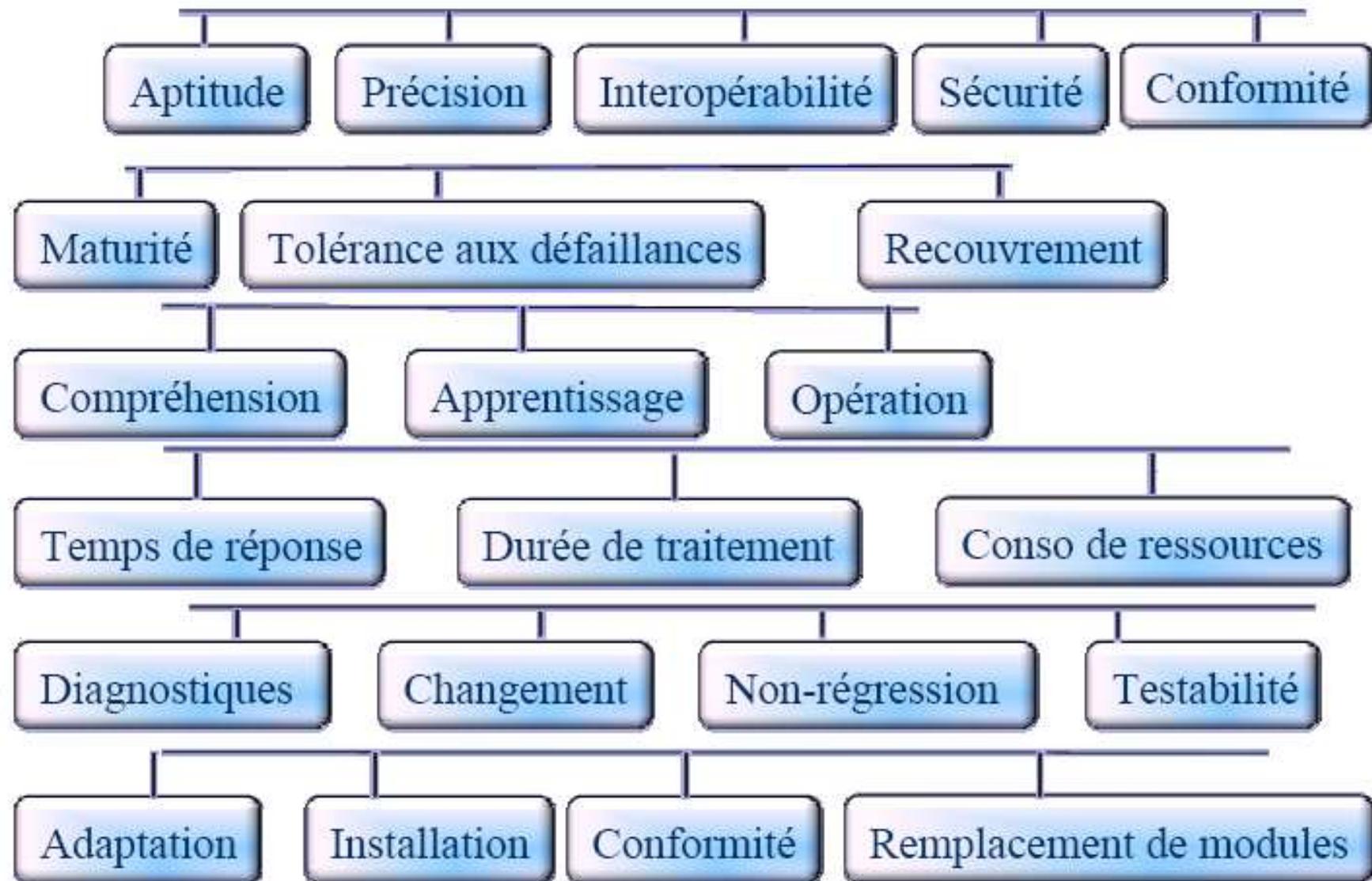
*Est-ce que le logiciel
requiert un
dimensionnement
rentable et proportionnée
de la plate-forme
d'hébergement
en regard des
autres exigences ?*

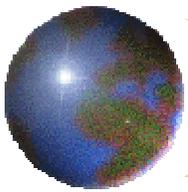
*Est-ce que le logiciel
requiert peu d'effort
pour son évolution
par rapport aux
nouveaux besoins ?*

*Est-ce que le logiciel
requiert peu d'effort
à l'utilisation ?*



Sous-caractéristiques





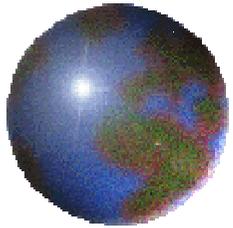
Bilan sur les normes

✚ Points positifs

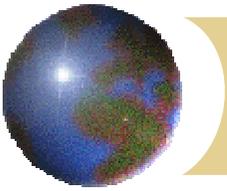
- ▣ Implication de tout le personnel
- ▣ Valorisation du professionnalisme des personnes

✚ Points négatifs

- ▣ Documentation excessive
- ▣ Pas d'innovation
- ▣ Echec si :
 - peu d'implication du management
 - Rédaction des procédures sans les acteurs
 - Communication insuffisante
 - Précipitation dans le lancement

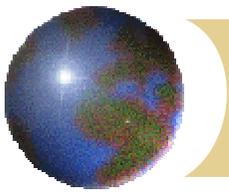


5. Qualité du logiciel



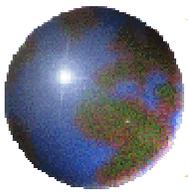
Introduction

- ✿ **Spécificités du produit-logiciel**
 - le logiciel est un produit complexe
 - beaucoup de modules
 - beaucoup d'opérations différentes possibles
 - le logiciel est un produit obscur
 - défauts non visibles à l'œil nu
 - leur détection nécessite des tests

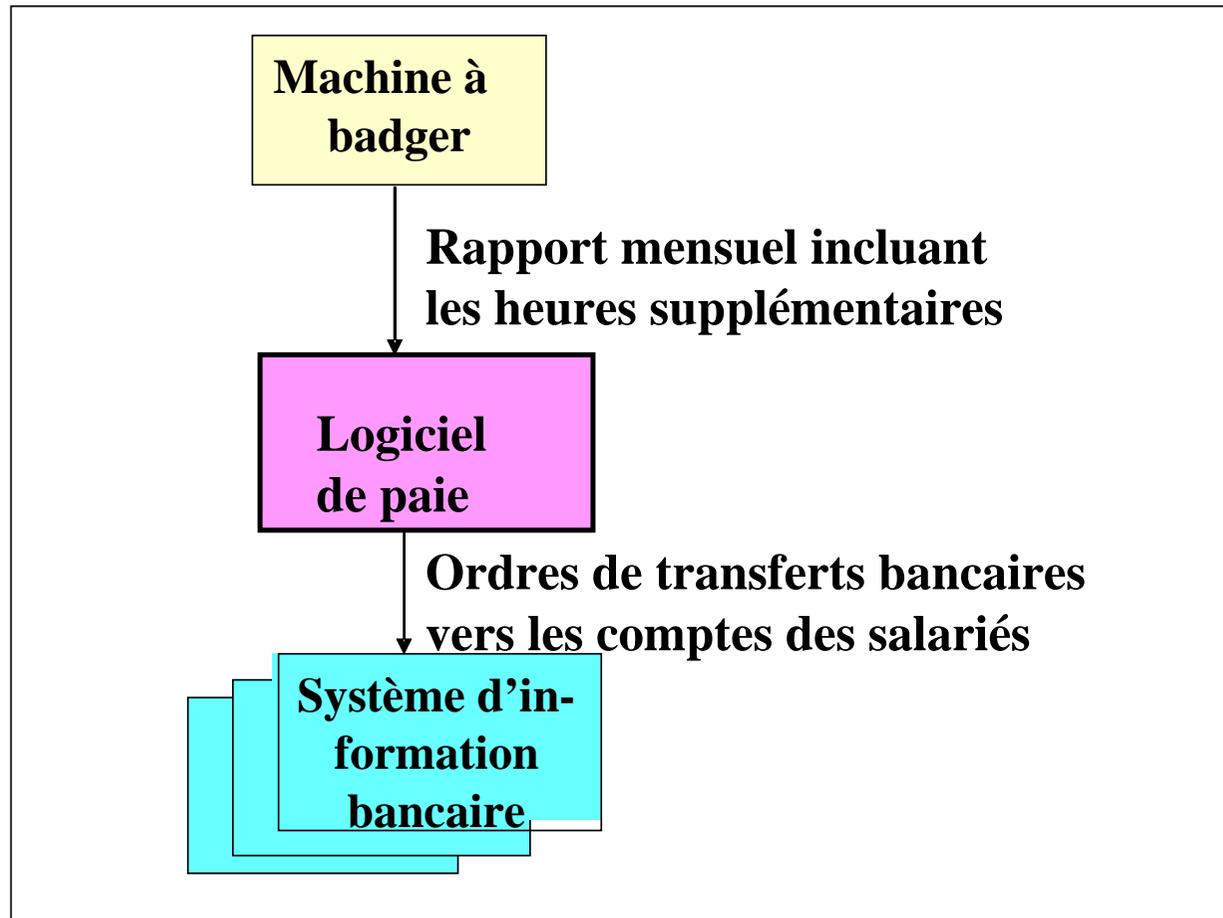


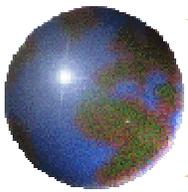
La qualité du logiciel

- ✿ Spécificités de l'environnement de production
 - ❑ contrat coût-délai-fonctionnalités
 - ❑ coopération étroite avec le client
 - ❑ travail d'équipe nécessitant une coopération et une coordination interne et externe
 - ❑ interfaces avec d'autres logiciels
 - ❑ taux important de rotation des équipes
 - ❑ nécessité de maintenance du logiciel pendant plusieurs années

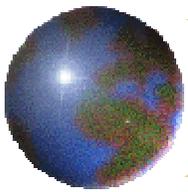


Exemple : logiciel de paie



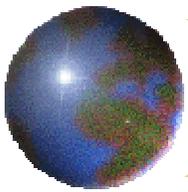


- ✚ Définition IEEE : Programmes informatiques, procédures, documentation associée et données nécessaires au lancement du système
- ✚ Une erreur dans le programme
 - peut déclencher un fonctionnement incorrect
 - qui peut entraîner un échec du système



Causes des erreurs

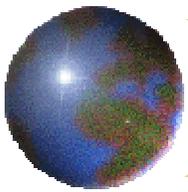
- ⊕ Définition incorrecte des besoins
- ⊕ Mauvaise communication entre le demandeur et le développeur
- ⊕ Modification délibérée du programme
- ⊕ Erreurs de logique
- ⊕ Erreurs de codage
- ⊕ Tests insuffisants
- ⊕ Erreurs dans les procédures



Qualité du logiciel

✚ Définition IEEE :

- ✚ degré auquel un système, un composant ou un processus satisfait les besoins spécifiés
- ✚ degré auquel un système, un composant ou un processus satisfait les attentes du client ou les besoins des utilisateurs

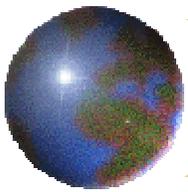


Qualité du logiciel

✚ Autre définition (Pressman, 2000) :

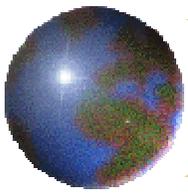
✚ conformité :

- aux besoins fonctionnels établis de façon explicite
- aux performances
- aux standards de développement demandés,
- aux caractéristiques implicites attendues de tous les logiciels développés de façon professionnelle



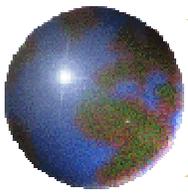
Des erreurs logicielles lourdes de conséquences ou non

- ⊕ La sonde Mariner devait passer à 5000 km de Vénus : elle s'est perdue à 5000000 de km à cause d'une erreur de programmation : un point remplacé par une virgule
- ⊕ Un appareil d'irradiation thérapeutique a provoqué la mort de 2 personnes et l'irradiation de 4 autres à cause d'un « bug »
- ⊕ Un navire de guerre anglais a été coulé par un Exocet français pendant la guerre des Malouines : plusieurs centaines de morts. La cause : le vaisseau anglais n'a pas activé ses défenses, parce que l'Exocet n'était pas répertorié dans les missiles ennemis
- ⊕ Inondation de la vallée du Colorado en 1983 à cause d'une erreur de modélisation dans le logiciel qui calcule le temps d'ouverture du barrage
- ⊕ En 1996, lors du 1er lancement d'Ariane V, elle explose en vol. La cause : réutilisation d'un logiciel d'Ariane IV qui vérifiait l'inclinaison de la fusée, mais le moteur étant plus puissant, il a jugé l'inclinaison non conforme à son plan de tir et a provoqué l'ordre d'autodestruction : 1/2 milliard de dollars
- ⊕ Saisie pour une dette impayée de 1 centime de franc : arrondi mal maîtrisé
- ⊕ Une centenaire (106 ans) convoquée à l'école : bug de l'an 2000
- ⊕ Une amende de 91500 \$ pour une cassette vidéo non restituée au bout de cent ans : bug de l'an 2000



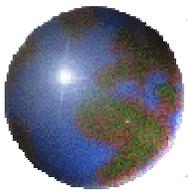
Facteurs de qualité du logiciel

- ✚ Permettent d'exprimer les exigences de qualité
- ✚ 3 classes
 - ▣ facteurs liés au fonctionnement du logiciel
 - ▣ facteurs liés à l'évolution du logiciel
 - ▣ facteurs liés à l'adaptabilité du logiciel



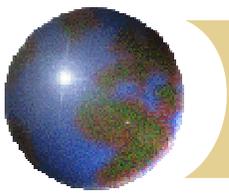
Modèle de McCall

- ✚ Facteurs de fonctionnement
 - ▣ exactitude, fiabilité, efficacité, intégrité, utilisabilité
- ✚ Facteurs d'évolution
 - ▣ maintenabilité, flexibilité, testabilité
- ✚ Facteurs d'adaptabilité
 - ▣ portabilité, réutilisabilité, interopérabilité



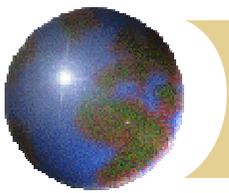
Autres facteurs

- Modèle d'Evans et Marciniak
 - vérifiabilité et extensibilité
- Modèle de Deutsch et Willis
 - sécurité, manageabilité, survivabilité



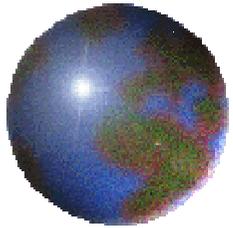
Facteurs de qualité interne selon Boehm*

Portabilité	Facilité à être porté sur de nouveaux environnements (matériels et/ou logiciels)
Traçabilité	Capacité à suivre un élément du cahier des charges jusqu'au composant logiciel
Vérifiabilité	Facilité de préparation des recettes
Intégrité	Aptitude d'un logiciel à protéger contre des modifications non autorisées
Fiabilité	Capacité à fonctionner sans tomber en panne
Documentation	Respect de standards
Testabilité	Facilité à localiser les défauts
Modularité	Degré d'organisation en modules
Interopérabilité	Aptitude à échanger avec d'autres systèmes
Compréhensibilité	Facilité à comprendre, notamment la documentation
Modifiabilité	Facilité à modifier
Clarté	Lisibilité et respect des conventions de réalisation

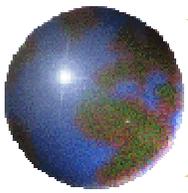


Facteurs de qualité externe selon Boehm*

Validité/conformité	Aptitude à réaliser les tâches définies dans les spécifications
Robustesse	Aptitude à fonctionner même dans des conditions anormales
Extensibilité/flexibilité	Facilité d'adaptation à un changement de spécification
Réutilisabilité	Aptitude d'un logiciel à être réutilisé en tout ou partie
Compatibilité	Aptitude du logiciel à pouvoir être combiné avec d'autres
Efficacité	Aptitude du logiciel à bien utiliser les ressources matérielles (temps CPU, mémoire, etc.)
Maintenabilité	Aptitude d'un logiciel à être facilement modifié
Facilité d'utilisation	Facilité d'apprentissage ou de préparation des données, de rattrapage en cas d'erreur d'utilisation, etc.
Économie	Adaptation par rapport aux moyens mis en oeuvre
Justesse	Aptitude du logiciel à fournir des résultats corrects
Elasticité	Capacité à minimiser l'effort en cas d'évolution
Généralité	Satisfaction du logiciel dans des environnements multiples



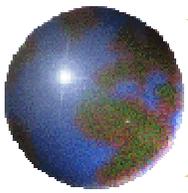
Les métriques de la qualité logicielle



Métrique de qualité logicielle

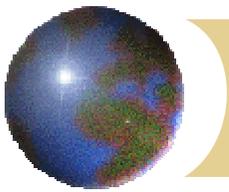
✚ Définitions IEEE :

- ✚ Mesure quantitative du degré auquel un élément tend vers un facteur de qualité
- ✚ Une fonction dont les entrées sont les données du logiciel et dont la sortie est une valeur numérique qui peut être interprétée comme le degré auquel un élément tend vers un facteur de qualité



Qu'est ce qu'une bonne métrique ?

- **Pertinente**
- **Valide**
- **Fiable**
- **Complète**
- **Mutuellement exclusive (une métrique = un facteur)**
- **Simple**
- **Ne nécessite pas la collecte de données supplémentaires**
- **Non sujette aux interventions des parties concernées**



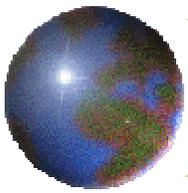
Classification des métriques de qualité

Par phase du cycle de vie

- ⊕ Métrique de processus
- ⊕ Métrique de produit

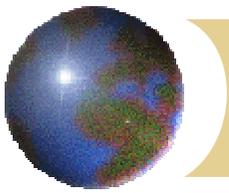
Par sujet

- ⊕ Qualité
- ⊕ Délai
- ⊕ Efficacité (des corrections, de la maintenance, des services)
- ⊕ Productivité



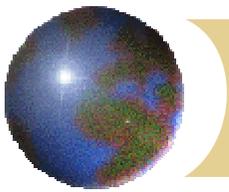
Mesures de la taille du logiciel

- ❖ **KLOC — milliers de lignes de code.**
- ❖ **Nb de points de fonction (NFP) —** mesures de la “quantité” de développement sur la base des fonctionnalités du logiciel



Les catégories de métriques de processus

- ❖ **Métrique de qualité du processus**
 - ❑ Métrique de densité d'erreur
 - ❑ Métrique de sévérité d'erreur
- ❖ **Métrique de délai du processus**
- ❖ **Métrique d'efficacité de la suppression d'erreur**
- ❖ **Métrique de productivité du processus**



Métriques de densité d'erreur

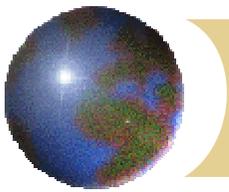
Code	Nom	Formule de calcul
CED	Densité d'erreur de codage	$CED = \frac{NCE}{KLOC}$
DED	Densité d'erreur de développement	$DED = \frac{NDE}{KLOC}$
WCED	Densité d'erreur de codage pondérée	$WCDE = \frac{WCE}{KLOC}$
WDED	Densité d'erreur de développement pondérée	$WDED = \frac{WDE}{KLOC}$
WCEF	Erreurs de codage par point de fonction	$WCEF = \frac{WCE}{NFP}$
WDEF	Erreurs de développement par point de fonction	$WDEF = \frac{WDE}{NFP}$

NCE = Nb d'erreurs de codage détectées pendant l'inspection du code ou pendant les tests

NDE = Nb total d'erreurs détectées (codage et logique) pendant le développement

WCED = Nb d'erreurs de codage détectées pendant l'inspection du code ou pendant les tests et pondérées par degré de sévérité

WDED = Nb total d'erreurs détectées pendant le développement et pondérées par sévérité



Métriques de sévérité d'erreur

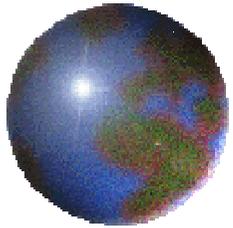
Code	Nom	Formule de calcul
ASCE	Sévérité moyenne des erreurs de codage	$\text{ASCE} = \frac{\text{WCE}}{\text{NCE}}$
ASDE	Sévérité moyenne des erreurs de développement	$\text{ASDE} = \frac{\text{WDE}}{\text{NDE}}$

NCE = Nb d'erreurs de codage détectées pendant l'inspection du code ou pendant les tests

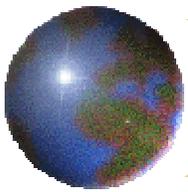
NDE = Nb total d'erreurs détectées (codage et logique) pendant le développement

WCE = Nb d'erreurs de codage détectées pendant l'inspection du code ou pendant les tests et pondérées par degré de sévérité

WDE = Nb total d'erreurs détectées pendant le développement et pondérées par sévérité

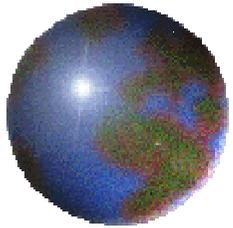


6. Qualité du processus logiciel

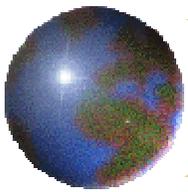


L'assurance qualité en avant-projet

- ✚ Revue de contrat
- ✚ Plan qualité



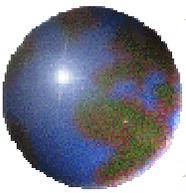
L'assurance qualité



Assurance qualité du logiciel

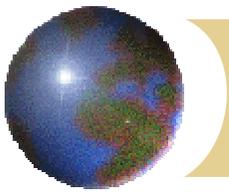
✚ Définition IEEE :

- ✚ schéma planifié et systématique de toutes les actions à conduire pour obtenir la confiance dans le fait qu'un produit est conforme aux exigences techniques établies
- ✚ ensemble d'activités d'évaluation du processus par lequel les produits sont fabriqués



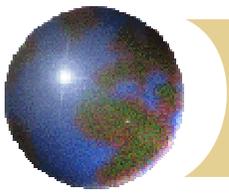
Assurance qualité du logiciel

- ✚ Autre définition (~ ISO 9000) :
 - ✚ ensemble systématique et planifié d'actions nécessaires pour fournir une confiance adéquate dans le fait que le processus de développement ou de maintenance du logiciel est conforme aux besoins fonctionnels et managériaux, respectant le planning et les contraintes budgétaires



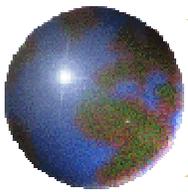
Objectifs de l'assurance qualité du développement de logiciel

- ❖ Assurer un niveau acceptable de confiance dans la conformité du logiciel aux spécifications fonctionnelles
- ❖ Assurer un niveau acceptable de confiance dans la conformité du logiciel au planning et au budget
- ❖ Assurer les activités d'amélioration du développement et d'assurance qualité



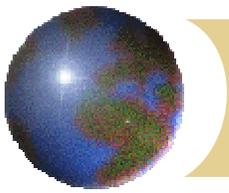
Objectifs de l'assurance qualité de la maintenance logicielle

- ❖ Assurer un niveau acceptable de confiance dans la conformité de la maintenance aux spécifications fonctionnelles
- ❖ Assurer un niveau acceptable de confiance dans la conformité de la maintenance au planning et au budget
- ❖ Assurer les activités d'amélioration de la maintenance et d'assurance qualité



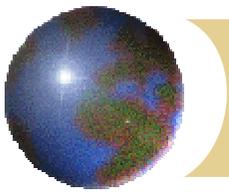
L'assurance qualité pendant le projet

- ⊕ Revues
- ⊕ Opinions d'expert
- ⊕ Test logiciel
- ⊕ Maintenance corrective, adaptative et évolutive
 - ⊕ Revue du contrat de maintenance
 - ⊕ Plan de maintenance
- ⊕ Assurance qualité dans l'implication de personnes extérieures (contrat)



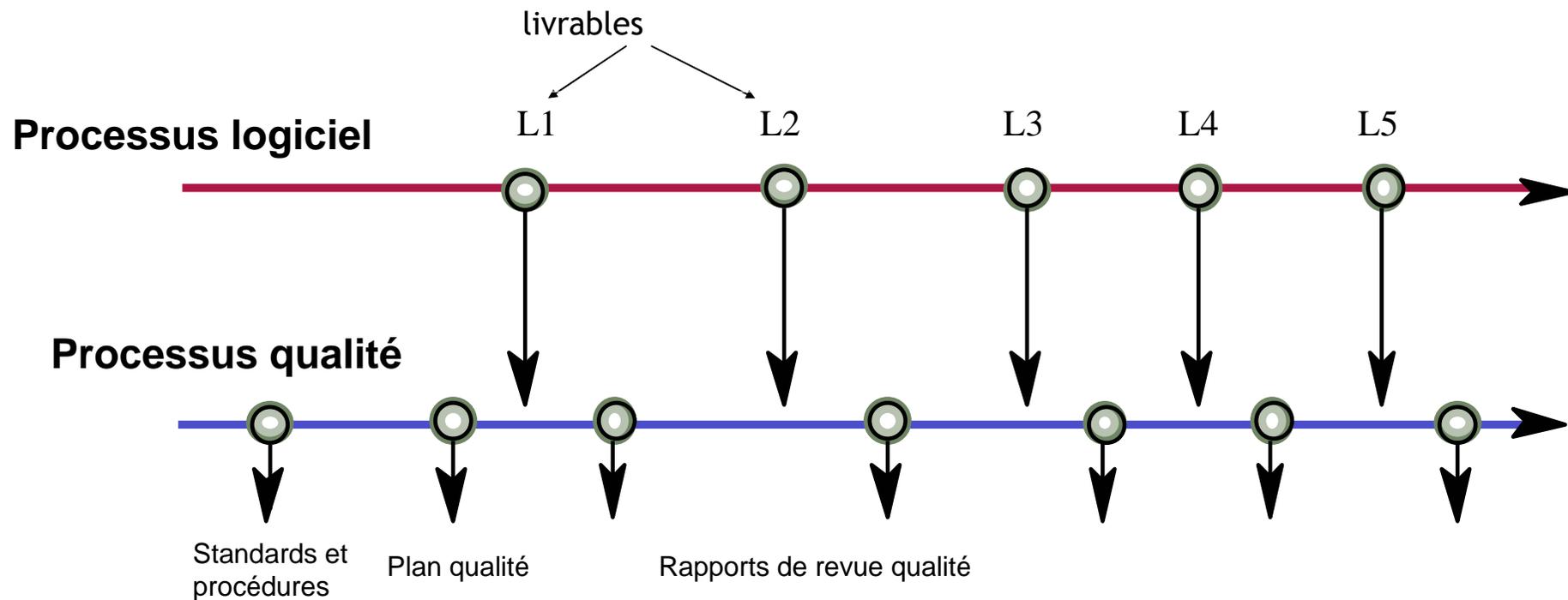
Infrastructure de prévention et d'amélioration

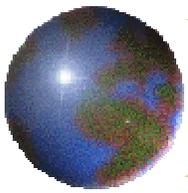
- ❖ Procédures et « checklists »
- ❖ Formation et certification des équipes
- ❖ Actions préventives et correctives
- ❖ Gestion de configuration
- ❖ Contrôle de la documentation



L'assurance qualité dans la gestion de projet

- Contrôle de l'avancement du projet
- Métriques de qualité du logiciel
- Suivi des coûts

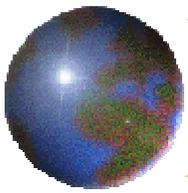




Les standards de l'assurance qualité

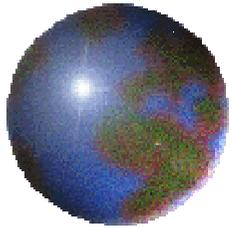
✚ Permettent :

- ✚ de se référer à la connaissance professionnelle internationale
- ✚ L'amélioration de la coordination avec les systèmes qualité des autres organisations
- ✚ L'évaluation et la mesure objectives des réalisations de l'organisation

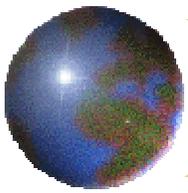


Les éléments humains de l'assurance qualité

- ✚ Importance du management
- ✚ Cellule qualité
- ✚ Comités qualité
- ✚ Forums qualité

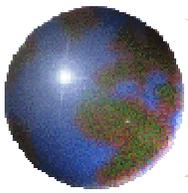


La revue de contrat



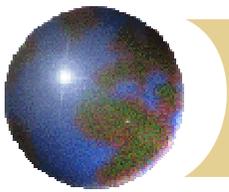
Les différents types de contrat

- ✚ Réponse à un appel d'offres
- ✚ Commande d'un client
- ✚ Demande interne
- ✚ Etc.



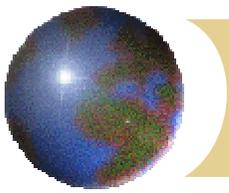
Les étapes de la revue de contrat

- ✚ Revue de la demande
- ✚ Revue de la réponse



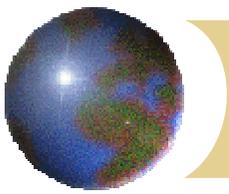
Revue de la demande

- S'assurer que :
 - Les demandes du client sont claires et documentées
 - Les approches alternatives ont été examinées
 - Les aspects formels de la relation client-fournisseur sont spécifiés
 - Les risques liés au développement sont identifiés
 - Les ressources et le délai nécessaires sont estimés de façon adéquate
 - La capacité du client à respecter ses engagements est examinée
 - Les conditions de participation d'autres partenaires ou sous-traitants sont définies



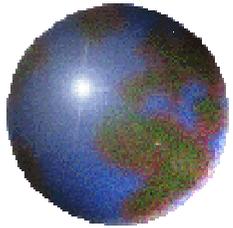
Revue de la réponse

- ❖ Pas de problème non clarifié dans la proposition
- ❖ Toutes les interprétations possibles de la demande sont considérées
- ❖ Il n'y a pas eu de changement depuis la rédaction de la demande (adéquation demande-réponse)

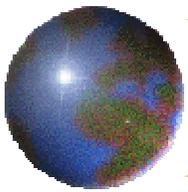


Cas particulier des projets « internes »

<i>Problème</i>	<i>Conséquence pour le "client"</i>	<i>Conséquence pour le développeur</i>
Définition insuffisante des besoins	Ecart entre l'attendu et le livré Insatisfaction du client	Changement constant des demandes du client
Pas d'estimation des ressources	Attentes irréalistes par rapport à la faisabilité du projet	Dérive des coûts Frictions dues aux demandes de "rallonges"
Planning inexistant ou succinct	Pas d'échéancier pour la livraison du produit	Pas de contrainte de délai Dérapage dans les délais et la qualité
Méconnaissance des risques	Client non préparé aux risques liés au projet et à leurs conséquences	Prise en compte tardive des problèmes

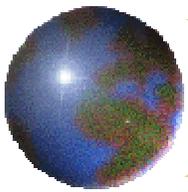


Le plan qualité du logiciel



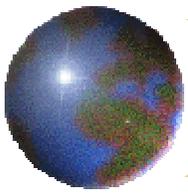
Objectifs du plan qualité

- ❖ Planifier la qualité permet de préparer les bases d'un projet réussi
- ❖ Inclut :
 - ❑ Planning
 - ❑ Estimation des ressources et du budget
 - ❑ Recrutement des équipes
 - ❑ Identification des risques
 - ❑ Mise en place des activités de l'assurance qualité
 - ❑ Fourniture au management des données nécessaires au contrôle du projet



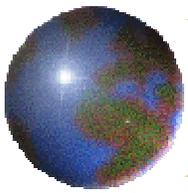
Les éléments d'un plan projet

- ⊕ Les livrables
- ⊕ Les interfaces
- ⊕ La méthodologie et les outils de développement
- ⊕ Les standards et les procédures
- ⊕ Les jalons
- ⊕ L'organisation du projet et la coordination avec les participants externes
- ⊕ Les risques liés et les actions de gestion des risques
- ⊕ Les moyens et méthodes de contrôle
- ⊕ Les estimations de coûts



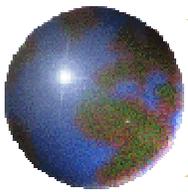
Les éléments d'un plan qualité

- ❖ Les objectifs de la qualité
- ❖ Les activités de revue
- ❖ Les tests
- ❖ Les recettes
- ❖ La gestion de configuration : outils, procédures, données, gestion des versions, etc.



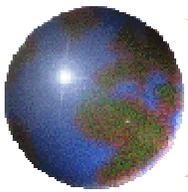
Risques liés au développement

- ❖ Dérapage du planning
- ❖ Fonctionnalités du système
- ❖ Sous traitance
- ❖ Prise en compte des besoins
- ❖ Utilisation des ressources et performance
- ❖ Gestion des ressources humaines

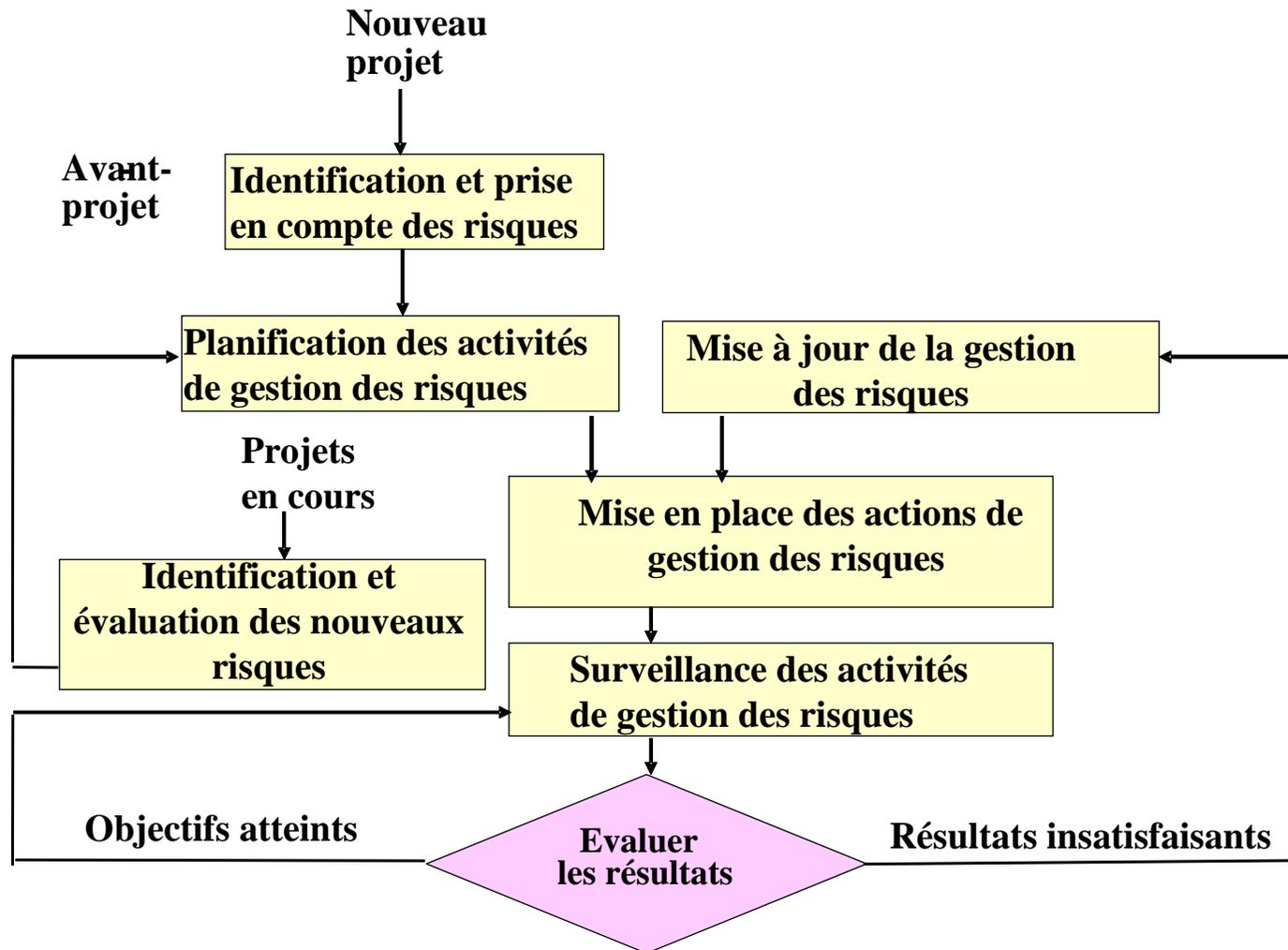


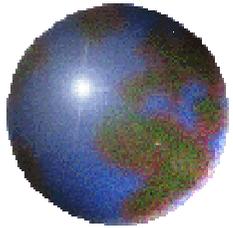
10 risques de Boehm et Ross

- ⊕ Développer des fonctionnalités non correctes
- ⊕ Délais et budgets irréalistes
- ⊕ Interfaces utilisateurs non adaptées
- ⊕ «Plaqué or »
- ⊕ Arrivée incessante de changements dans les demandes
- ⊕ Défauts dans les composants externes
- ⊕ Défauts dans les développements externes
- ⊕ Défaillance du personnel (« turnover »)
- ⊕ Performances système insuffisantes
- ⊕ Limites des possibilités techniques

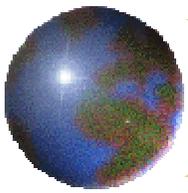


Le processus de gestion de risques



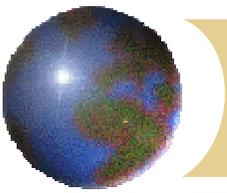


L'intégration de la qualité dans la gestion de projet

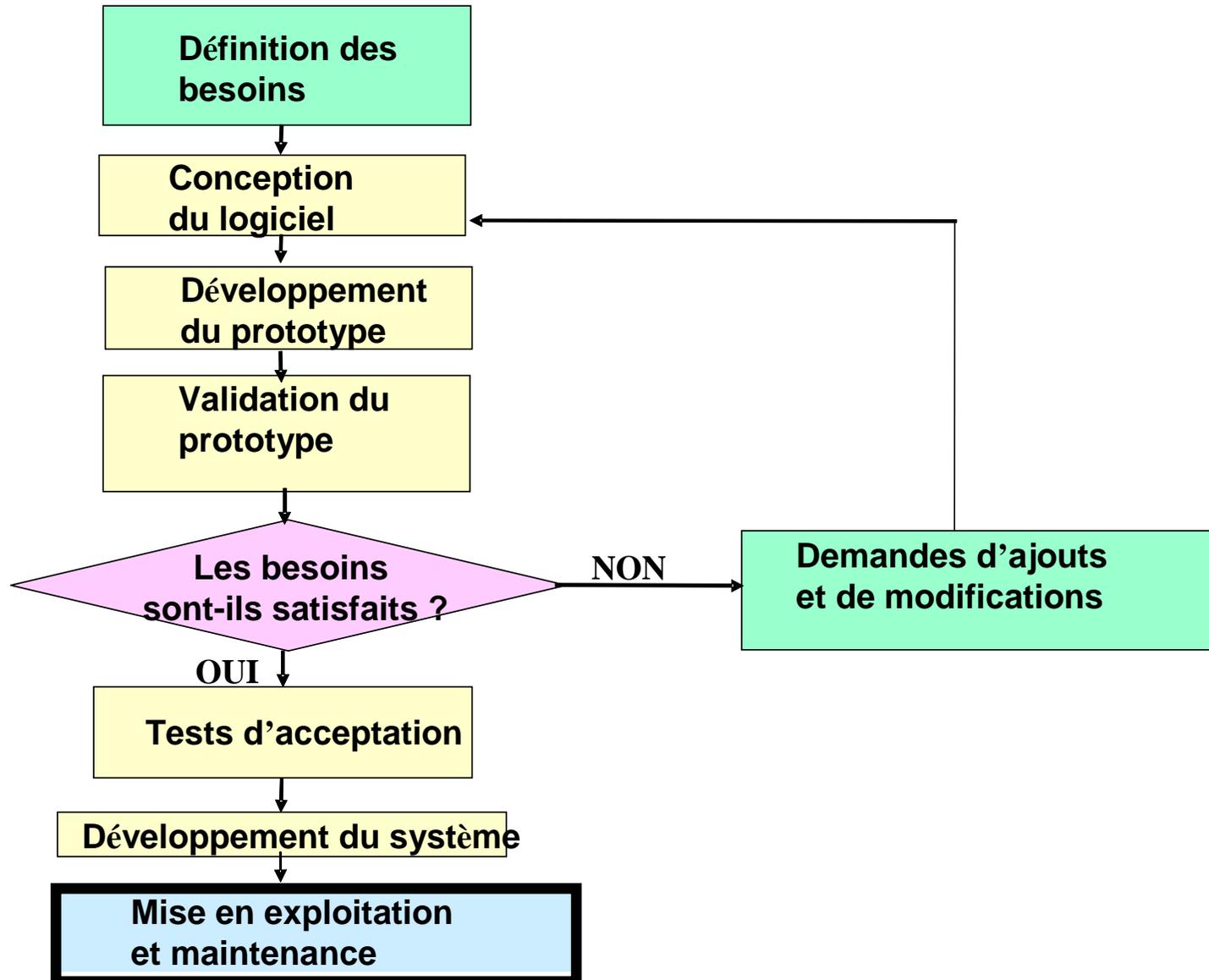


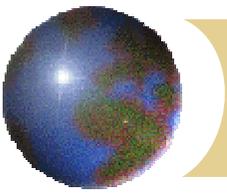
Méthodologies de gestion de projet

- ⊕ Cycle de vie du produit logiciel
- ⊕ Prototypage
- ⊕ Modèle en spirale

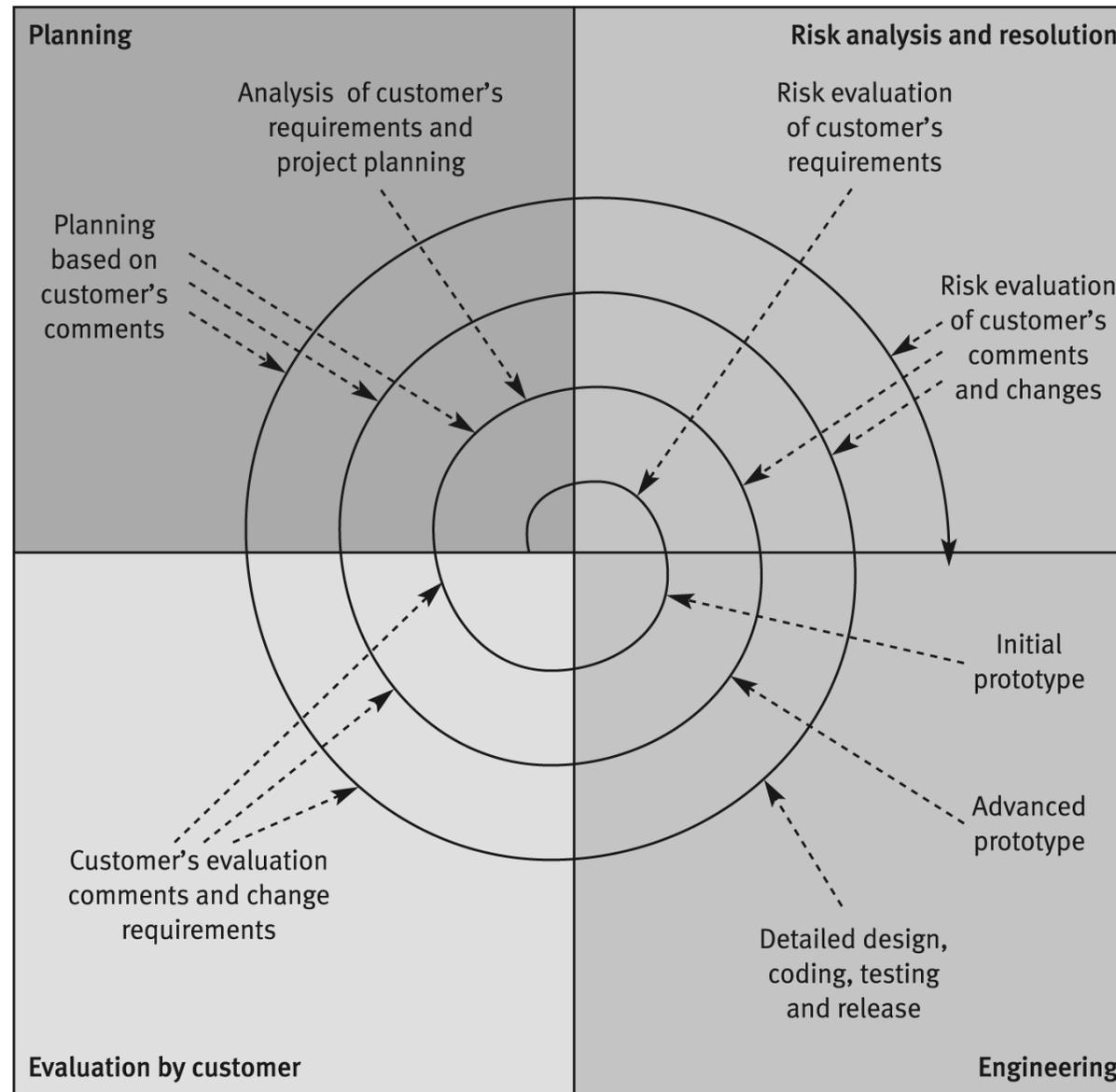


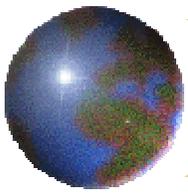
Prototypage



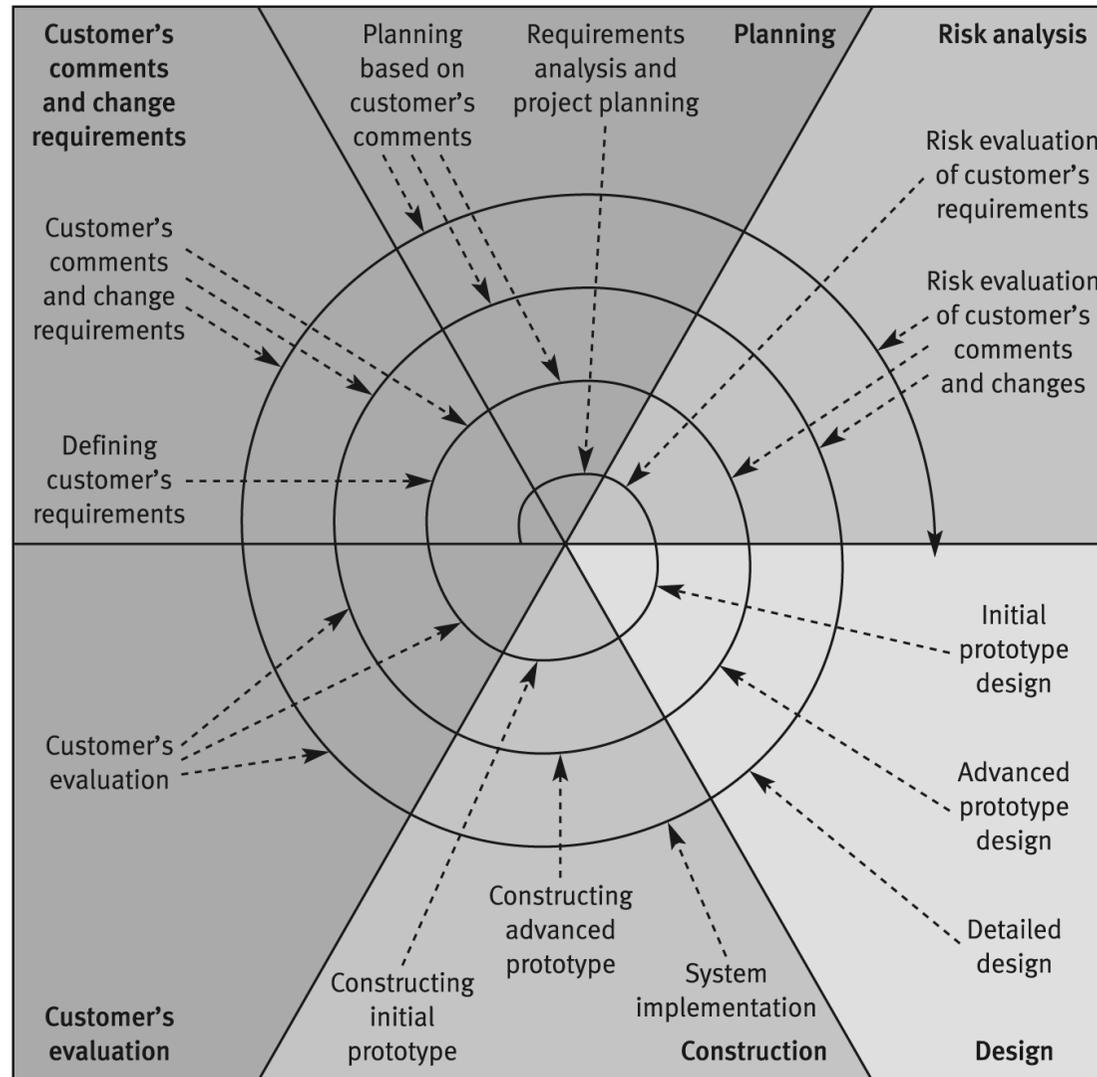


Modèle en spirale

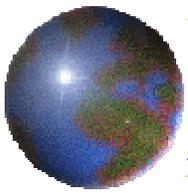




Modèle en spirale gagnant-gagnant

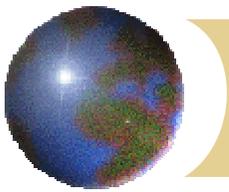


Source: After Boehm 1998 (© 1988 IEEE)



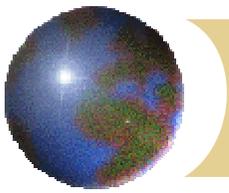
Facteurs influant sur le processus qualité

- Facteurs liés au projet
 - Taille du projet
 - Complexité technique du projet
 - Ajout de composants logiciels
 - Gravité des conséquences en cas d'échec du projet
- Facteurs liés à l'équipe
 - Qualification des membres
 - Proximité avec le projet et expérience dans le domaine
 - Disponibilité des équipes métiers
 - Ancienneté des individus et des équipes



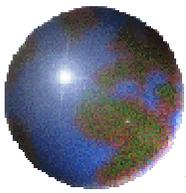
Vérification, validation et qualification

- ❊ Vérification : processus d'évaluation d'un système ou d'un composant pour déterminer si les livrables d'une phase du développement satisfont les conditions imposées au début de cette phase
 - ❏ « building the product right »
- ❊ Validation : processus d'évaluation d'un système ou d'un composant pendant ou à la fin du processus de développement pour déterminer s'il satisfait les spécifications
 - ❏ « building the right product »
- ❊ Qualification : processus utilisé pour déterminer si un système ou un composant est opérationnel



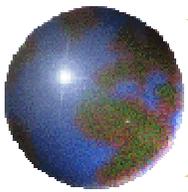
Modèle d'estimation des coûts liés aux défaut

- ✚ Résultats quantitatifs :
 - ▣ Efficacité du plan qualité dans l'élimination des défauts des logiciels
 - ▣ Coûts totaux d'élimination des défauts



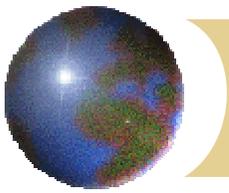
Origine des défauts vs coûts de leur élimination

Phase du projet	Pourcentage moyen de défauts générés	Coût moyen relatif d'élimination du défaut
Spécification des besoins	15%	1
Conception	35%	2.5
Codage unitaire	30%	6.5
Intégration	10%	16
Documentation	10%	16
Test du système	-----	40
Exploitation	-----	110



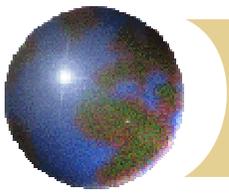
Efficacité de la détection des erreurs avec plan qualité

Activité d'assurance qualité	Efficacité de la détection des erreurs avec plan standard	Efficacité de la détection des erreurs avec plan détaillé
Revue de la spécification des besoins	50%	60%
Inspection de la conception	-----	70%
Revue de la conception	50%	60%
Inspection du code	-----	70%
Test du code	50%	40%
Tests d'intégration	50%	60%
Revue de la documentation	50%	60%
Test du système	50%	60%
Détection pendant l'exploitation	100%	100%

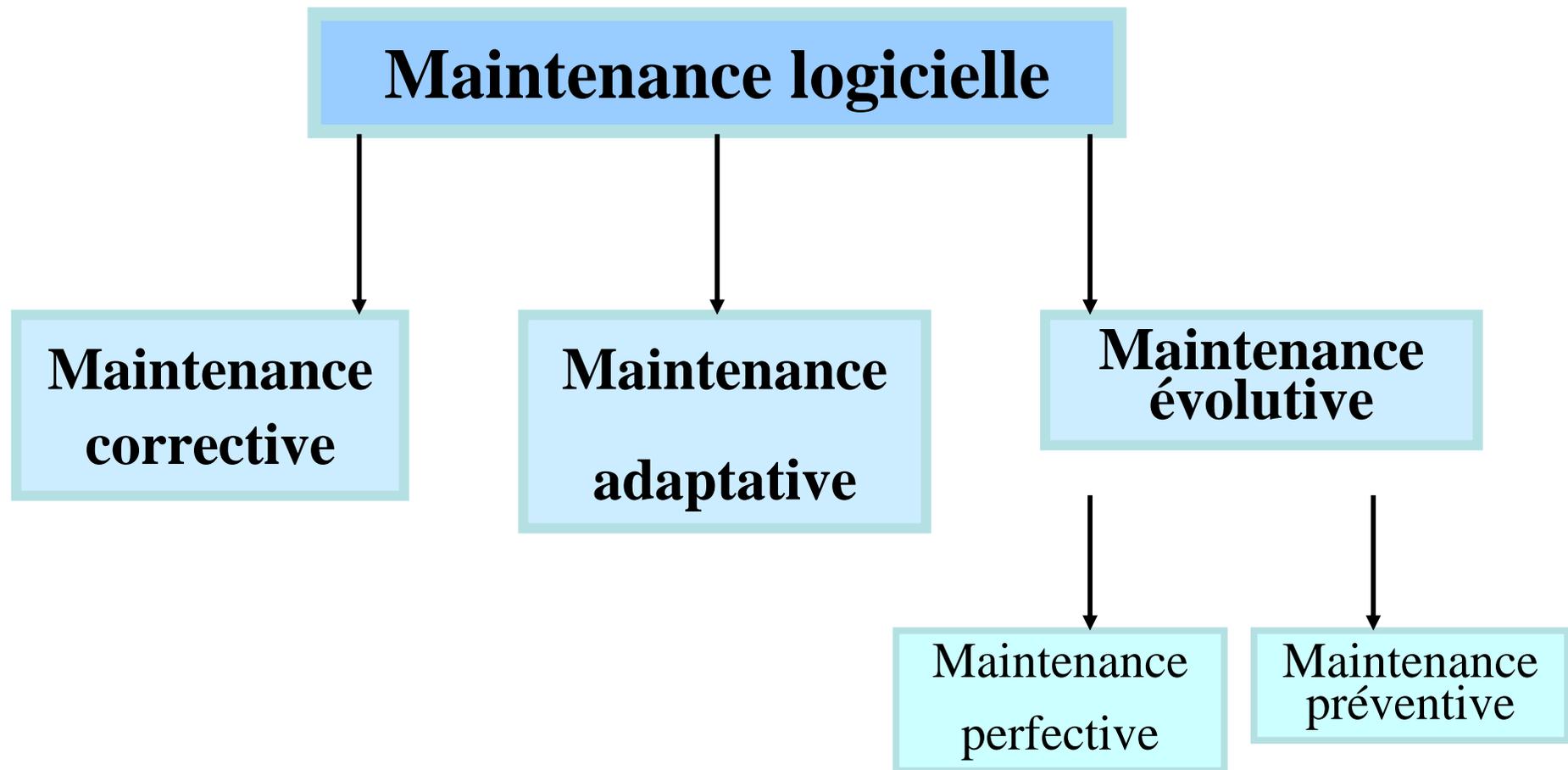


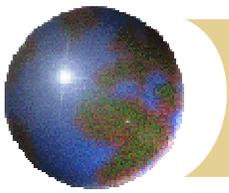
Effficacité de la suppression des erreurs avec plan qualité

Phase d'élimination des défauts	Efficacité de l'élimination	Coût relatif moyen de suppression d'un défaut				
		Phase d'origine du défaut				
		Besoins	Conception	Codage	Intégration	Doc.
Spécification des besoins	50%	1	---	---	---	---
Conception	50%	2.5	1	---	---	---
Codage unitaire	50%	6.5	2.6	1	---	---
Intégration	50%	16	6.4	2.5	1	
Documentation système	50%	16	6.4	2.5		1
Test système / Test d'acceptation	50%	40	16	6.2	2.5	1
Effectué par le client (après livraison)	100%	110	44	17	6.9	2.5



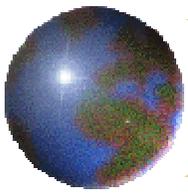
Maintenance logicielle





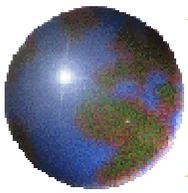
Facteurs qualité ayant un impact fort sur la maintenance logicielle

Facteur de qualité	Maintenance corrective	Maintenance adaptative	Maintenance évolutive
Exactitude des sorties	+++		
Exactitude de la documentation	+++	+++	+++
Exactitude de la qualification	+++	+++	+++
Fiabilité	+++		
Maintenabilité	+++	+++	+++
Flexibilité		+++	
Testabilité	+++		
Portabilité		+++	
Interopérabilité		+++	



Composants de l'assurance qualité : la revue de maintenance

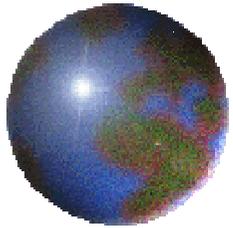
- ❖ **Clarification des besoins du "client"**
- ❖ **Analyse des alternatives de réalisation de la maintenance**
- ❖ **Revue de l'estimation des moyens requis**
- ❖ **Revue des services de maintenance fournis par la sous-traitance ou par le client**
- ❖ **Revue de l'estimation des coûts**



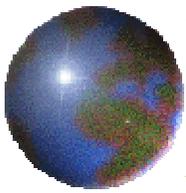
Composants de l'assurance qualité : le plan de maintenance

✦ Comprend :

- Une liste des services de maintenance faisant l'objet d'un contrat (clients internes ou externes)**
- Une description de l'organisation de l'équipe de maintenance**
- Une liste des outils de maintenance**
- Une liste des risques liés à la maintenance**
- Une liste des procédures de maintenance**
- Le budget**

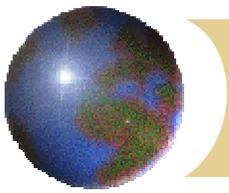


Les outils CASE et leur effet sur la qualité



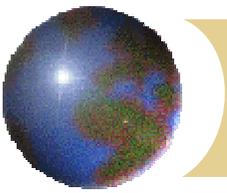
Degré de contribution des outils CASE à la qualité

Causes des erreurs	Degré de contribution
1. Mauvaise définition des besoins	Presque aucun
2. Mauvaise communication	Presque aucun
3. Déviations délibérées par rapport aux spécifications	++
4. Erreurs de logique	++
5. Erreurs de codage	+++
6. Décalage entre le codage et la documentation	+++
7. Insuffisances des tests	++
8. Interface utilisateur et erreurs dans les procédures	+
9. Erreurs de documentation	++



Contribution des outils CASE à la qualité de la maintenance

- La documentation générée et mise à jour par l'outil facilite l'identification des erreurs**
- Les contrôles de cohérence du CASE facilitent l'anticipation des erreurs**
- La correction dans les outils "lower CASE" ou CASE intégrés fournit une documentation automatique des corrections**



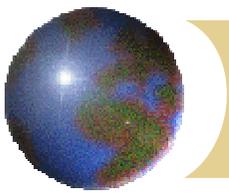
Métriques de délai

Code	Nom	Formule de calcul
TTO	Respect du planning	$\text{TTO} = \frac{\text{MSOT}}{\text{MS}}$
ADMC	Retard moyen par jalon	$\text{ADMC} = \frac{\text{TCDAM}}{\text{MS}}$

MSOT = Jalons terminés à temps

MS = Nb total de jalons.

TCDAM = Total des retards (jours, semaines, etc.) pour tous les jalons



Métriques d'efficacité de la correction d'erreur

Code	Nom	Formule de calcul
DERE	Efficacité de la suppression des erreurs	$\text{DERE} = \frac{\text{NDE}}{\text{NDE} + \text{NYF}}$
DWERE	Efficacité pondérée	$\text{DWERE} = \frac{\text{WDE}}{\text{WDE} + \text{WYF}}$

NCE = Nb d'erreurs de codage détectées pendant l'inspection du code ou pendant les tests

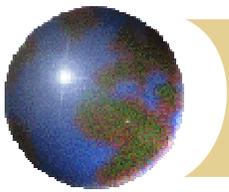
NDE = Nb total d'erreurs détectées (codage et logique) pendant le développement

WCE = Nb d'erreurs de codage détectées pendant l'inspection du code ou pendant les tests et pondérées par degré de sévérité

WDE = Nb total d'erreurs détectées pendant le développement et pondérées par sévérité

NYF = Nb d'échecs du logiciel détectés pendant un an de maintenance

WYF = Nb d'échecs du logiciel détectés pendant un an de maintenance pondérés par sévérité



Métriques de productivité du processus

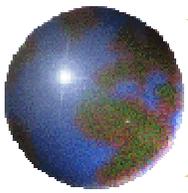
Code	Nom	Formule de calcul
DevP	Productivité du développement	$\text{DevP} = \frac{\text{KLOC}}{\text{DevH}}$
FDevP	Productivité du développement par PF	$\text{FDevP} = \frac{\text{NFP}}{\text{DevH}}$
Cre	Réutilisation de code	$\text{Cre} = \frac{\text{ReKLOC}}{\text{KLOC}}$
DocRe	Réutilisation de documentation	$\text{DocRe} = \frac{\text{ReDoc}}{\text{NDoc}}$

DevH = Nb d'heures total de développement.

ReKLOC = NB de milliers de lignes de code réutilisées.

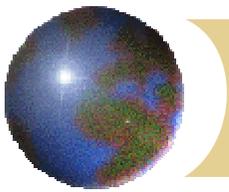
ReDoc = Nb de pages de documentation réutilisées.

NDoc = Nb de pages de documentation.



Les catégories de métriques de produits

- **Métrique de qualité du helpdesk (HD)**
 - Métrique de densité d'appel
 - Métrique de sévérité d'appel
 - Métrique de succès du HD
- **Métrique de productivité du HD**
- **Métrique d'efficacité du HD**
- **Métrique de qualité de la maintenance corrective**



Métriques de densité d'appels HD

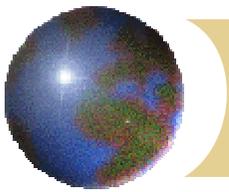
Code	Nom	Formule de calcul
HDD	Densité des appels HD	$\text{HDD} = \frac{\text{NHYC}}{\text{KLMC}}$
WHDD	Densité pondérée des appels HD	$\text{WHYC} = \frac{\text{WHYC}}{\text{KLMC}}$
WHDF	Densité pondérée par point de fonction	$\text{WHDF} = \frac{\text{WHYC}}{\text{NMFP}}$

NHYP = nb d'appels HD pendant une année.

KLMP = milliers de ligne de code à gérer.

WHYP = nb d'appels HD pendant une année pondérés par sévérité.

NMP = nb de points de fonction à gérer

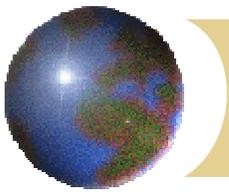


Métriques de sévérité d'appels HD

Code	Nom	Formule de calcul
ASHC	Sévérité moyenne des appels HD	$\text{ASHC} = \frac{\text{WHYC}}{\text{NHYC}}$

NHYC = Nb d'appels HD pendant un an

WHYC = Nb d'appels HD pendant un an pondérés par sévérité

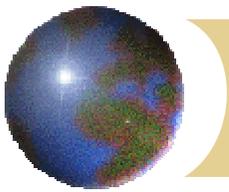


Métriques de succès du HD

Code	Nom	Formule de calcul
HDS	Succès du service de HD	$\text{HDS} = \frac{\text{NHYOT}}{\text{NHYC}}$

NHYOT = Nb d'appels HD traités dans les délais pendant un an.

NHYC = Nb d'appels HD reçus pendant un an.



Métriques de productivité et d'efficacité du HD

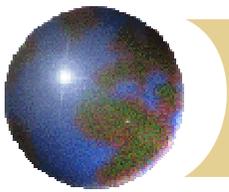
Code	Nom	Formule de calcul
HDP	Productivité du HD	$\text{HDP} = \frac{\text{KLNC}}{\text{HDYH}}$
FHDP	Productivité du HD par point de fonction	$\text{FHDP} = \frac{\text{NMFP}}{\text{HDYH}}$
HDE	Efficacité du HD	$\text{HDE} = \frac{\text{NHYC}}{\text{HDYH}}$

HDYH = Nb d'heures total de HD.

KLNC = Nb de KLOC gérées

NMFP = Nb de points de fonction gérés.

NHYC = Nb d'appels HD pendant un an.



Métriques de densité d'échec du logiciel

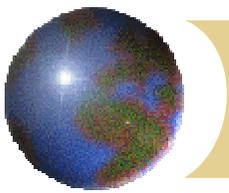
Code	Nom	Formule de calcul
SSFD	Densité d'échec du logiciel	$\text{SSFD} = \frac{\text{NYF}}{\text{KLMC}}$
WSSFD	Densité d'échec du logiciel pondérée	$\text{WSSFD} = \frac{\text{WYF}}{\text{KLMC}}$
WSSFF	Densité d'échec par point de fonction	$\text{WSSFF} = \frac{\text{WYF}}{\text{NMFP}}$

NYF = nb d'échecs logiciels détectés pendant un an

WYF = nb d'échecs logiciels détectés pondérés pendant un an

NMFP = nb de points de fonction

KLMC = nb de milliers de lignes de code.

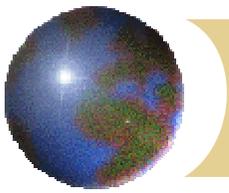


Métriques de sévérité d'échec du logiciel

Code	Nom	Formule de calcul
ASSSF	Sévérité moyenne des échecs du logiciel	$\text{ASSSF} = \frac{\text{WYF}}{\text{NYF}}$

NYF = nb d'échecs pendant un an.

WYF = nb d'échecs pondérés par sévérité pendant un an.

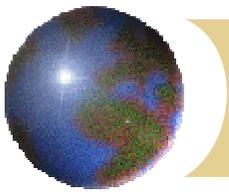


Métriques d'échec du service de maintenance

Code	Nom	Formule de calcul
MRepF	Métrique d'appels d'échecs répétés	$\text{MRepF} = \frac{\text{RepYF}}{\text{NYF}}$

NYF = nb d'échecs pendant un an

RepYF = nb d'échecs répétés pendant un an



Métriques de disponibilité du logiciel

Code	Nom	Formule de calcul
FA	Disponibilité pleine	$FA = \frac{NYSerH - NYFH}{NYSerH}$
VitA	Disponibilité vitale	$VitA = \frac{NYSerH - NYVitFH}{NYSerH}$
TUA	Indisponibilité totale	$TUA = \frac{NYTFH}{NYSerH}$

NYSerH = Nb d'heures de fonctionnement du logiciel par an

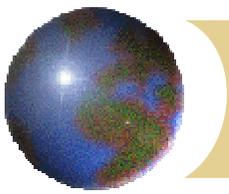
NYFH = Nb d'heures où au moins une fonction n'est pas disponible

NYVitFH = Nb d'heures où au moins une fonction vitale n'est pas disponible

NYTFH = Nb d'heures d'arrêt total dû à un échec

$NYFH \geq NYVitFH \geq NYTFH$.

$1 - TUA \geq VitA \geq FA$



Métriques de productivité et d'efficacité de la maintenance corrective

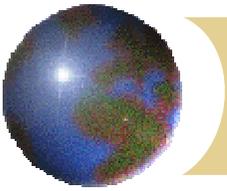
Code	Nom	Formule de calcul
CMaiP	Productivité de la maintenance corrective	$\text{CMaiP} = \frac{\text{KLMC}}{\text{CMaiYH}}$
FCMP	Productivité de la maintenance corrective par point de fonction	$\text{FCMP} = \frac{\text{NMFP}}{\text{CMaiYH}}$
CMaiE	Efficacité de la maintenance corrective	$\text{CMaiE} = \frac{\text{NYF}}{\text{CMaiYH}}$

CMaiYH = Nb d'heures de maintenance corrective par an

NYF = Nb d'échecs du logiciel par an

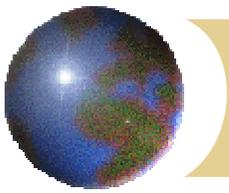
NMFP = Nb de points de fonction

KLMC = Milliers de lignes de code



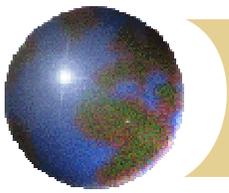
Limitations des métriques

- **Contraintes budgétaires**
- **Facteurs humains : réticence à l'auto-évaluation**
- **Validité : incertitude sur les données**
- **Difficultés liées aux paramètres (voir suite)**



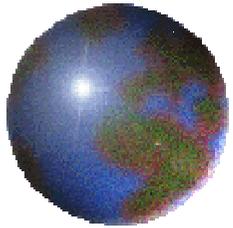
Facteurs affectant les paramètres des métriques utilisées pour le processus de développement

- a. Style de programmation (**KLOC**).
- b. Volume des commentaires (**KLOC**).
- c. Complexité du logiciel (**KLOC, NCE**).
- d. Pourcentage de code réutilisé (**NDE, NCE**).
- e. Professionnalisme dans la détection d'erreurs (**NCE**).
- f. Styles de rapports de développement (**NDE, NCE**).

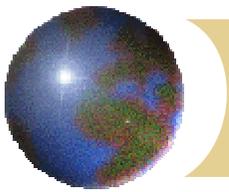


Facteurs affectant les paramètres des métriques utilisées pour le processus de maintenance

- a. Qualité du logiciel et de sa documentation **(NYF, NHYC)**.
- b. Style de programmation et volume des commentaires **(KLMC)**.
- c. Complexité du logiciel **(NYF)**.
- d. Pourcentage de code réutilisé **(NYF)**.
- e. Nb d'installations, nb d'utilisateurs, niveau d'utilisation **(NHYC, NYF)**.



7. Conclusion



Conclusion

- ➊ Domaine en pleine expansion
- ➋ Lié à d'autres phénomènes :
 - ▣ Certification
 - ▣ Master data management
- ➌ Pléthore d'outils :
 - ▣ Lesquels utiliser ?
 - ▣ Comment choisir ?
 - ▣ Comment les mettre entre les mains des non informaticiens ?
- ➍ Des domaines spécifiques
 - ▣ Qualité des données géographiques
 - ▣ Qualité des informations sur Internet
 - ▣ Qualité dans le domaine médical