

Bonsoir !

INF755 Méthodes d'analyse et de conception

Hiver 2018

Séance-12

Chargé de cours: Alain Cardinal

Plan de la séance-12

- **Retour séance-12**
- Maintenance des SI
 - **Problèmes** avec la maintenance
 - **Définition** de la maintenance
 - **Perception** de la maintenance
 - **Environnement** de la maintenance

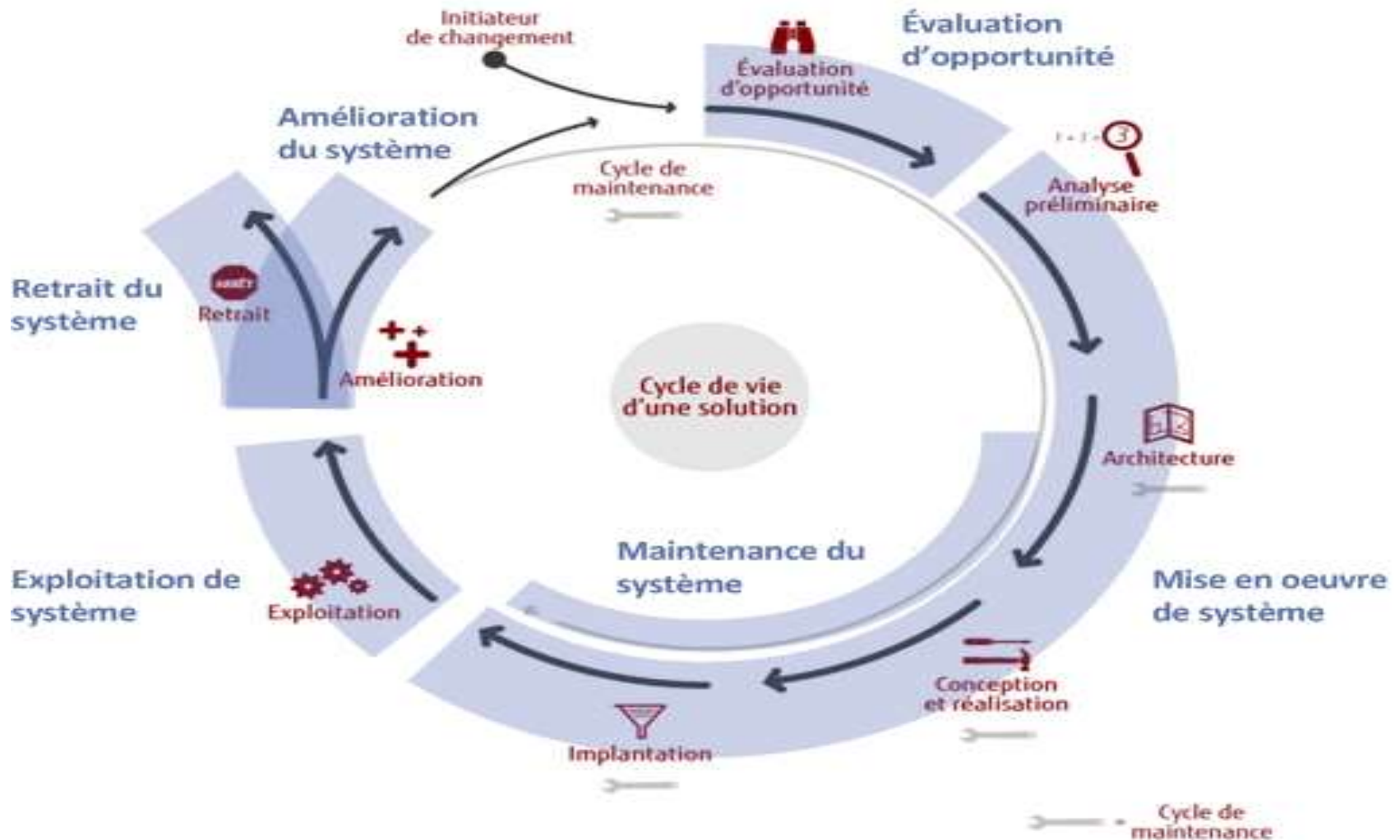
Retour séance-11

- **SAP**
 - **ERP**
 - **ASAP**
 - **ABAP**

Plan de la séance-12

- **Maintenance des SI**
 - **Problèmes** avec la maintenance
 - **Définition** de la maintenance
 - **Perception** de la maintenance
 - **Environnement** de la maintenance

Étapes du cycle de vie



Problème avec la maintenance vue par le client

- Coût élevé de la maintenance
- Service de maintenance lent
 - Ressources vs. @Backlog@
- Ne comprend pas comment le service de maintenance est priorisé et assigné

Opinion d'experts

- La maintenance logicielle souffre plus d'un **manque de gestion** que d'un simple problème de connaissances techniques
 - On ne **reconnait** pas la maintenance
 - On ne **structure** pas la maintenance
 - On ne **mesure** pas la maintenance

Collecte inadéquate des données

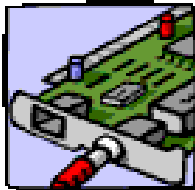
- Les gestionnaires de la maintenance ont de la difficulté à expliquer leurs coûts.
 - **Que peuvent 'ils utiliser ?**
- Les gestionnaires de la maintenance ont de la difficulté à expliquer la valeur ajoutée de la maintenance.



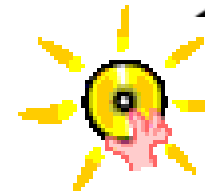
Maintenance matérielle VS Maintenance logicielle?

- ⊙ Remplacer une composante
- ⊙ On pèse sur le bouton pour vérifier

⊙ ?



VS.



Maintenance matérielle VS Maintenance logicielle?

- Le matériel se dégrade sans maintenance
- Le logiciel se dégrade lorsqu'il est maintenu!!!

Maintenance matérielle VS Maintenance logicielle?

Matériel

- ⊙ Photocopier Repair \$15/hr
- ⊙ PC Repair \$20/hr
- ⊙ Camera Repair \$20/hr
- ⊙ TV Repair \$20/hr
- ⊙ Audio Repair \$25/hr



Logiciel

- ⊙ Web maintenance \$45/hr
- ⊙ Cobol Maintenance \$55/hr
- ⊙ Database maintenance \$65/hr
- ⊙ OS maintenance \$75/hr
- ⊙ ERP maintenance \$130/hr



Plan de la séance-12

- Retour séance-11
- Maintenance des SI
 - **Problèmes** avec la maintenance
 - **Définition** de la maintenance
 - **Perception de la maintenance**
 - **Environnement** de la maintenance

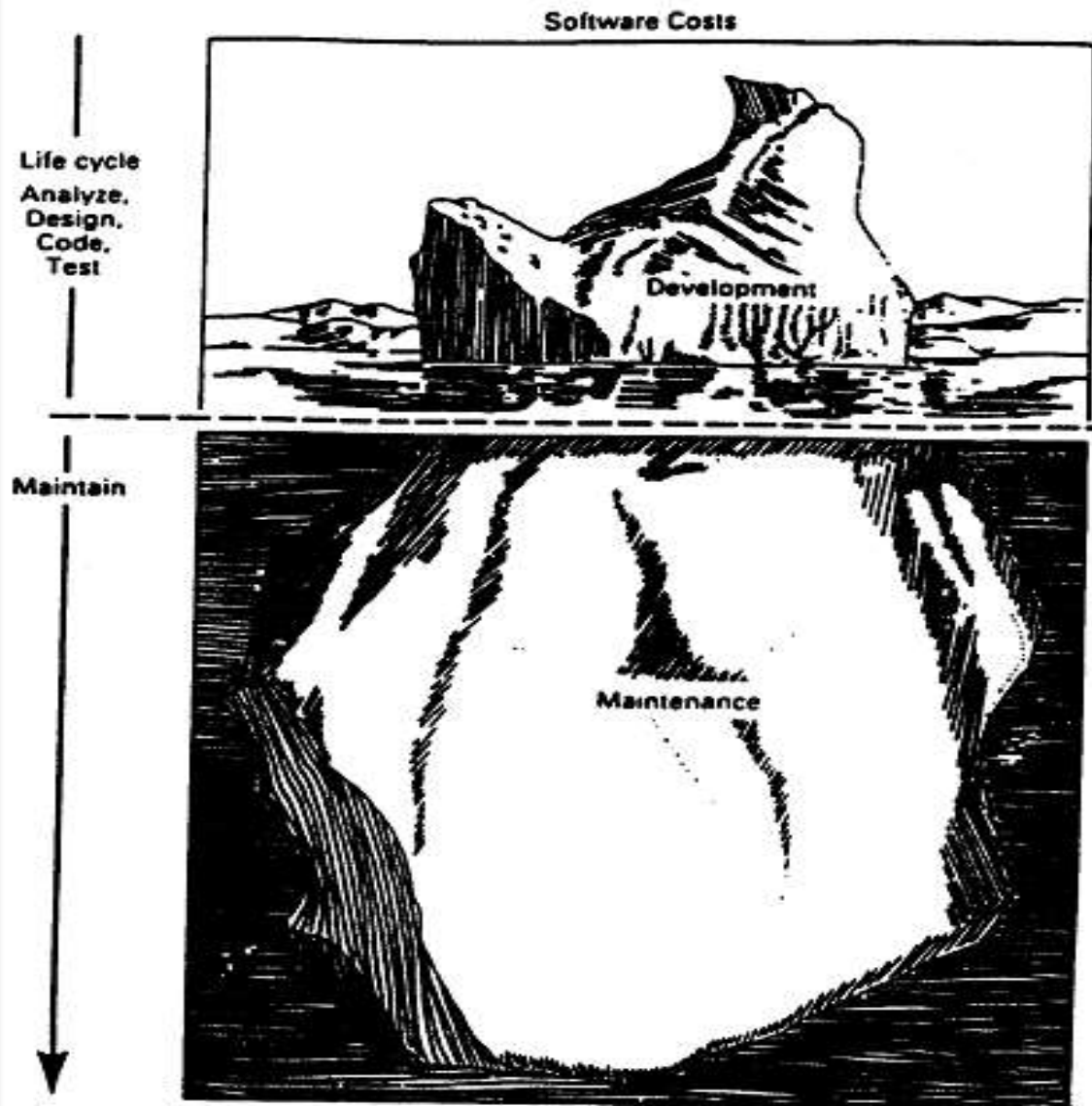


Figure 1.1 Software development represents the tip of the iceberg. The majority of the software budget and the majority of the software professional's time are spent on software maintenance, not software development.

Perception

- Coût élevé et mauvais service
 - Le client paie beaucoup
 - Il ne voit pas la valeur pour son argent
 - Pannes en production
 - Erreurs logicielles
 - Longs délais d'attente
 - Coût élevé associé à un petit changement

SLA flous

- La personne à la maintenance doit mettre fin à tous les autres travaux lorsqu'une défaillance en production se produit
 - Perception : l'entretien ne se fait pas selon la priorité des utilisateurs
- Les utilisateurs ont tendance à classer toutes les demandes en tant que hautes
- Les utilisateurs envoient beaucoup de demandes (comité de gestion des demandes!)

- Entente de service

Est-ce que ces perceptions sont réelles?

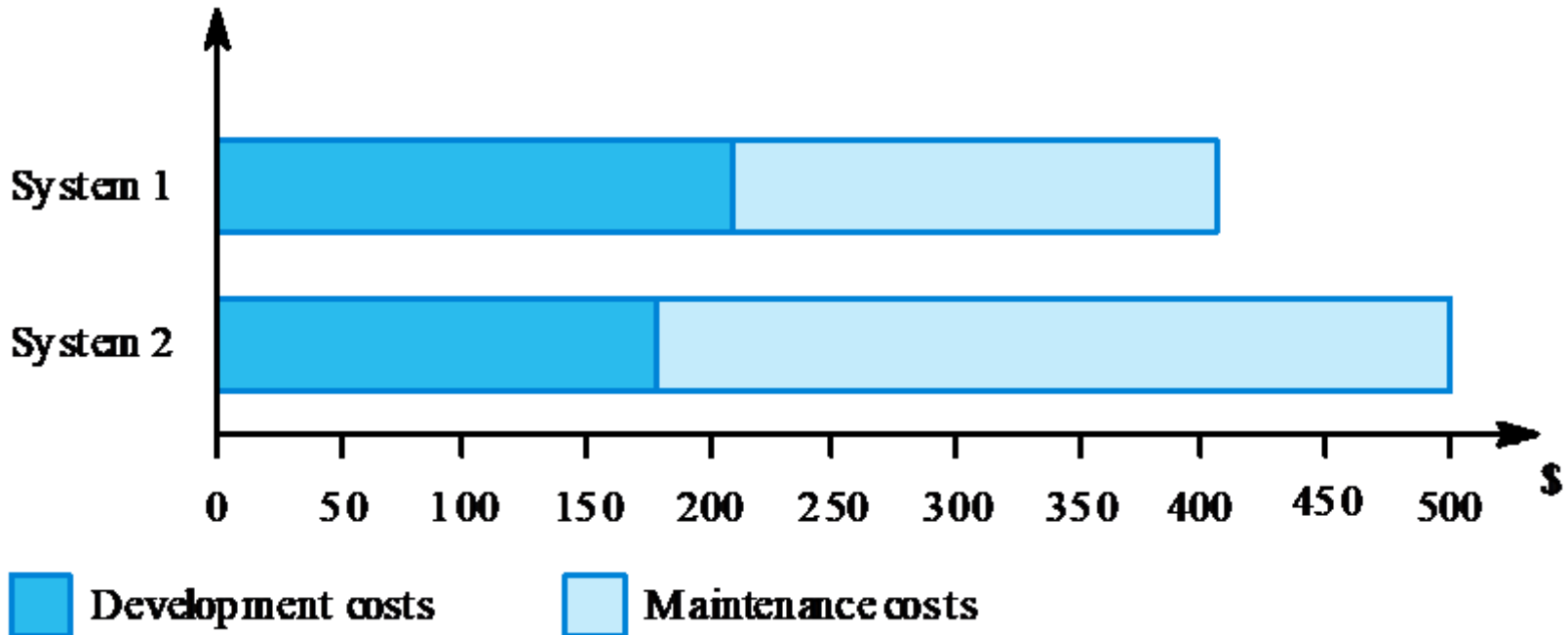
- Plusieurs réponses possibles!
 - Maturité du mainteneur
 - Maturité du client
 - Quel type de logiciel

Temps de développement vs. Temps de maintenance

- Un projet de développement typique prend entre 1 et 2 ans
- La maintenance requiert un temps additionnel de 5 à 6 ans
- Règle du 80-20 (Pareto):
 - 20 % de l'effort dans le développement
 - 80 % de l'effort dans la maintenance.

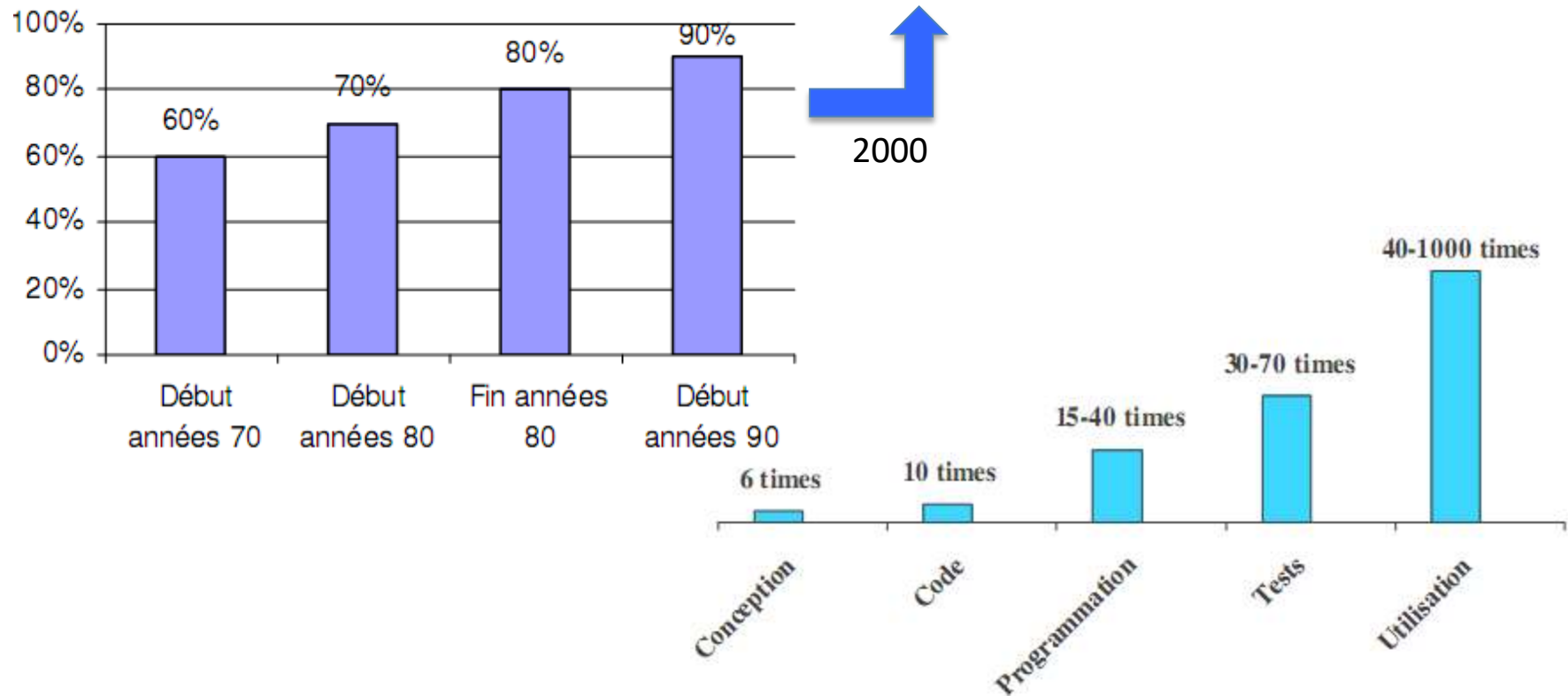
Mauvaise perception

Oui, la maintenance est le coût le plus élevé!!



Les coûts de la maintenance

Coûts de la maintenance



Coût de la maintenance

- Habituellement plus élevé que le développement
 - 2 à 100 fois plus, dépendant de l'application
- Touché par des facteurs techniques et non techniques
- Augmente au fur et à mesure que le logiciel est maintenu.
 - La maintenance dégrade la structure du logiciel, rendant la maintenance plus difficile.
- Les vieux systèmes peuvent avoir des coûts plus élevés
 - vieux langages, compilateurs

Fausse perception des coûts

- Le service de maintenance n'est pas clair lorsque:
 - Les parties et les processus sont immatures (pas de SLA);
 - Symptômes
 - priorités confuses, délais, coûts non justifiés
- Oui, la main d'œuvre est chère, mais lorsqu'elle est utilisée pour ajouter de nouvelles fonctionnalités...

Perception interne

- Logiciel très mal conçu et programmé
 - Conception douteuse
- Absence de documentation
- Pas de système de test automatisé

Réalité de mainteneur

- Logiciels large et complexe avec peu de documentation;
- Doit être changé rapidement sans interrompre le service;

Les lois de Lehman

- **Changement continue**

- Un logiciel doit continuellement être adapté. Sinon la satisfaction du client diminuera.
- Ces changements se font **sous l'effet de la pression** de l'environnement (les utilisateurs) dans lequel le système évolue et par l'écart qui existe entre les caractéristiques du système et les besoins du domaine.

- **Complexité croissante**

- La complexité du logiciel augmente à chaque évolution à moins que cette évolution soit faite pour maintenir ou diminuer cette complexité. Les interactions et les dépendances augmentent, le couplage croît et le risque d'erreur croît.
- Si beaucoup d'effort est consacré à éviter la croissance de la complexité, alors ce sont moins de ressources et de temps disponibles pour la transformation du système.

Lois de Lehman

- **Croissance continue**
 - Le contenu fonctionnel d'un logiciel doit continuellement croître afin de satisfaire les besoins des utilisateurs durant toute sa vie utile.
- **Qualité décroissante**
 - La qualité d'un logiciel va être perçue comme décroissante à moins d'une maintenance rigoureuse et adaptée aux changements dans l'environnement opérationnel.

Conclusion sur les lois

- **Observations:**
 - Les logiciels évoluent ou disparaissent
 - Un logiciel qui grossit implique une complexité qui va limiter sa capacité de croître.
- **Solutions:**
 - Nécessité de gérer la complexité
 - Revoir périodiquement le design
 - Activités de refactoring pour rajeunir le logiciel

Classification des problèmes selon Dr. Bennet

- **Problèmes d'alignement (A)**
- **Problèmes de processus (P)**
- **Problèmes techniques (T)**
 - (A) Et (P) sont des problèmes de management
 - (T) Sont des problèmes techniques
- Est-ce qu'il est vrai que la plupart des problèmes de maintenance sont des problèmes de management?

Le système est imposé

- **Imposé aux mainteneurs**
 - Il a souvent beaucoup de problèmes
- Un grand nombre de problèmes à résoudre
 - Un backlog qui vient avec
- Pas de support du management pour vous aider dans cette situation
 - Faites-le!

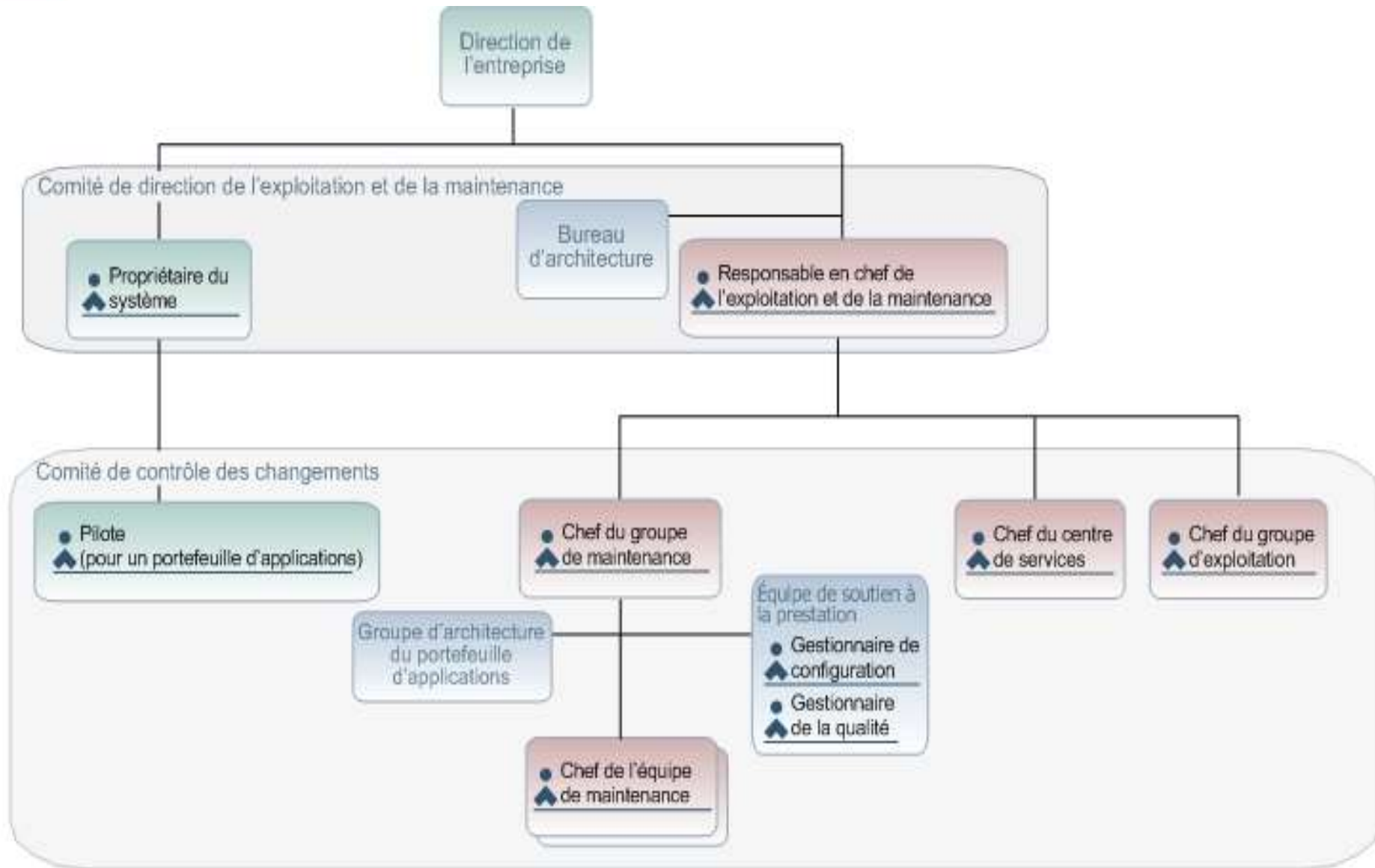
Le conflit



- Les clients ont besoin d'un système maintenant! Les attentes sont élevées.
- Les développeurs construisent rapidement (plus vite et en deçà du budget si possible)
- Les mainteneurs veulent un système sans défaut, venant avec une documentation pour la maintenance du système pour les prochaines années...
- La DDC???

- DDC

Comité de contrôle des changements

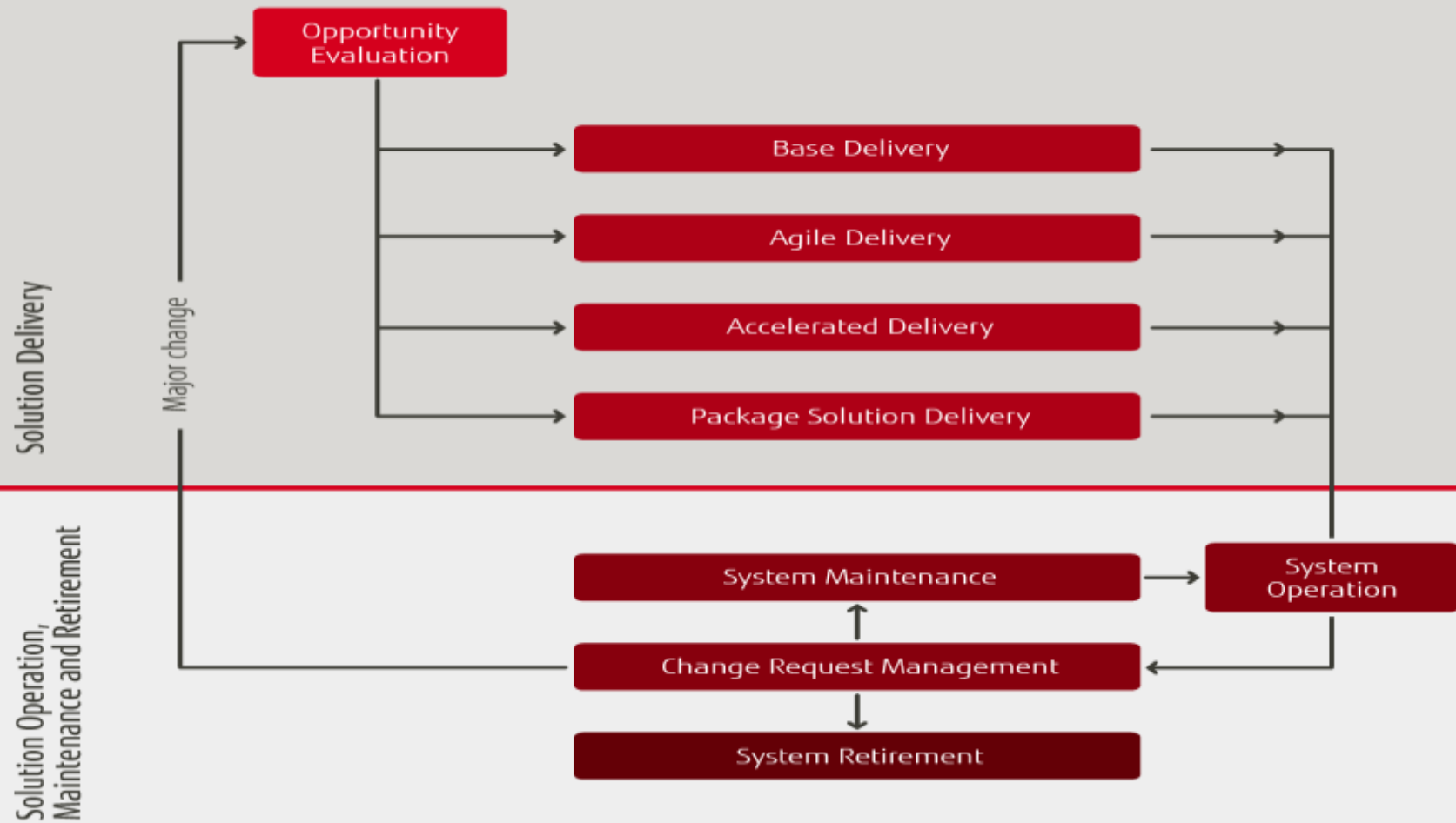


Trois perspectives

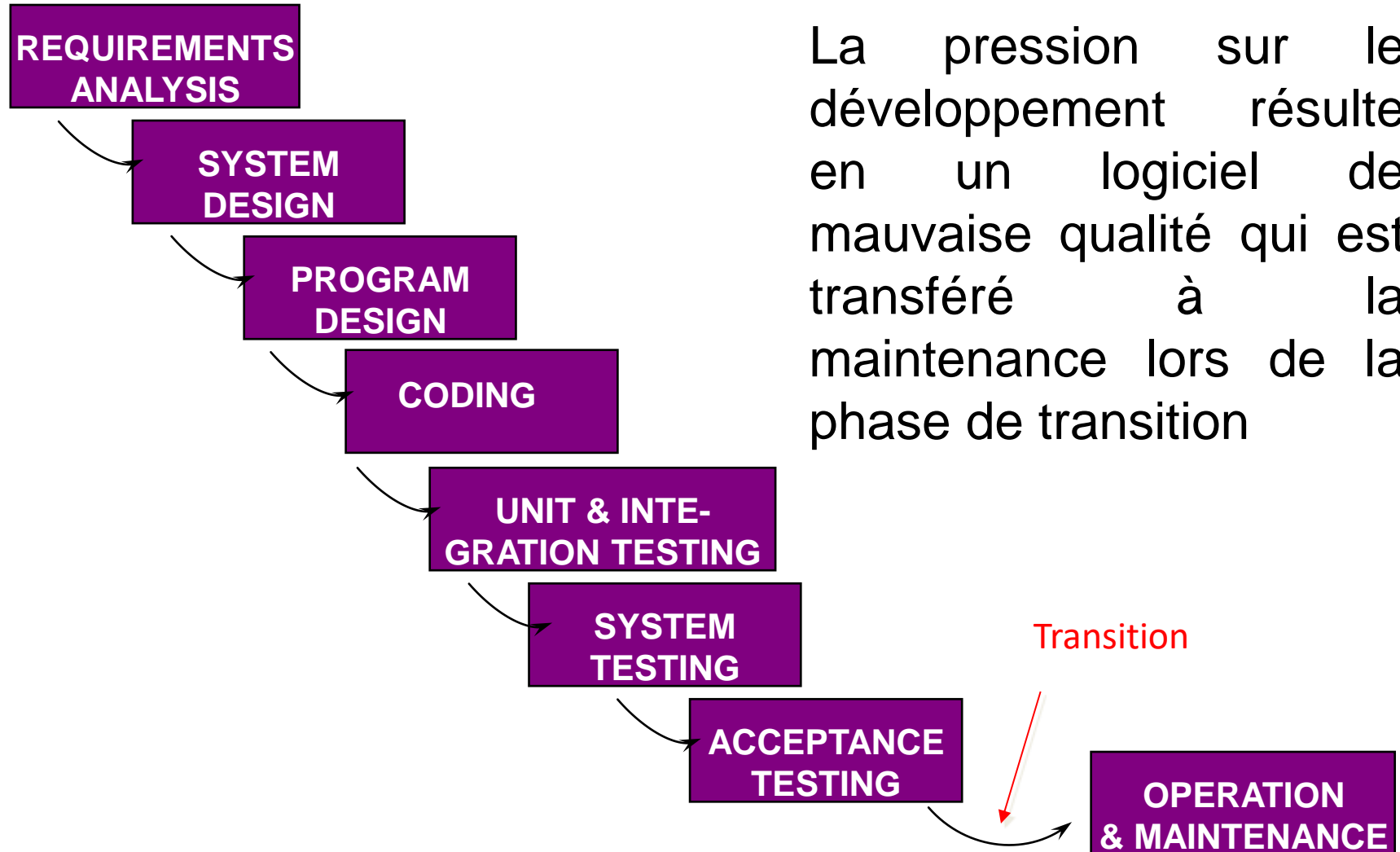
 Besoins  Ressources  Solution

Macroscopic solution ?

Solution Life Cycle



Points de pression



Coûts d'une faute

- Le coût de correction d'une faute croît avec le temps
- Si trouvée, combien elle coûte à réparer?
- Prenons un problème dans les exigences, mais que nous identifions plus tard...
 - 4x si trouver et corriger pendant le design
 - 10x si trouver et corriger pendant l'implémentation
 - 30 – 52 x si trouver...lors de l'intégration
 - 200 – 368 x si trouver ... lors de la maintenance

Processus immatures

- Âge moyen des codeurs
 - en inde: 26-27
 - En afrique du sud : 28-32
- Il y a toujours une grande proportion de développeurs/processus de développement immatures.
- Les processus matures permettent de créer des logiciels plus faciles à maintenir
- Ces logiciels arrivent souvent avec un gros backlog
 - Clients insatisfaits

Plan de la séance-12

- **Retour séance-11**
- **Maintenance des SI**
 - **Problèmes** avec la maintenance
 - **Définition** de la maintenance
 - **Perception** de la maintenance
 - **Environnement de la maintenance**

L'environnement de travail de la maintenance



Développeur



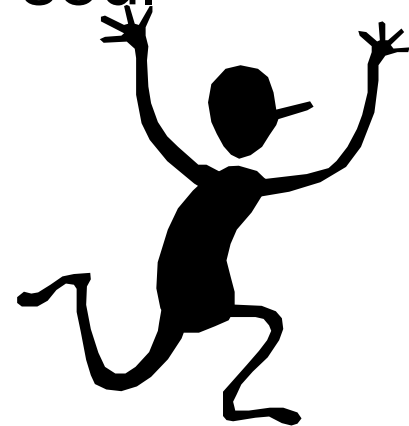
Bad



Mainteneur

Problème: Le morale

- **11.9 %** des problèmes pendant la maintenance sont le résultat d'un mauvais moral
- Pendant la maintenance, 8 % des problèmes proviennent d'un programmeur poussé dans trop de dossiers, l'empêchant de se concentrer suffisamment longtemps pour régler un seul problème.



Sari Lawrence Pfleeger

Inexpérience

- Qui fait la maintenance:
 - ⊙ 25 % des étudiants
 - ⊙ 61 % des programmeurs d'expériences (Swanson-Beath)
 - ⊙ Dans une autre étude
 - ⊙ 60 % à 80 % - personnes récemment engagées.

Qui fait la maintenance

- Parfois des développeurs:
 - Moins d'intérêts, il veulent quitter pour un meilleur travail (créatif)
 - Difficile d'embaucher
 - Problème de continuité (roulement)
 - Moins d'indépendance
 - Moins bonne qualité
 - Moins de transparence

Qui fait la maintenance

- Parfois des mainteneurs:
 - Grandes organisations
 - Plus d'indépendance
 - Qualité et transparence accrues
 - Compétition pour résoudre les problèmes entre développeurs et mainteneurs
 - Besoin d'établir la frontière entre le développement et la maintenance

Sommaires

- Point de vue divergent avec client/utilisateur
- Grande proportion de problèmes de management
- Les rapports de maintenance ne montrent pas réellement ce qui se passe.
- La complexité des logiciels est croissante
- Le coût de la maintenance est élevé et en croissance
- La mauvaise qualité du développement est toujours un problème (Agile...)
- La maintenance n'est pas un travail intéressant (morale)
- Manque de méthodologies/Outils/Formation

Processus de la maintenance

- Nécessité d'identifier le travail correspondant à une maintenance versus un projet de développement
- Une modification qui prend 5 mois n'est sûrement pas de la maintenance
- Le processus n'est pas le même

Catégoriser les billets

Définitions:

- **Haute**
 - Le problème est sérieux
 - Le client ne peut plus travailler
 - Il n'y a pas de voie de contournement connue
- **Élevée**
 - La majorité des problèmes des applications
 - Représente 80 % du volume
- **Basse**
 - Problèmes mineurs

<http://www.lgc.com/customersupport/supportservices/servicelevels/default.htm>

Résolution d'un billet

- **Résolution:** *L'intervalle de temps entre la demande de support et lorsque l'analyste technique trouve une solution et ferme le billet.*
 - Répondre à une question
 - Expliquer un processus
 - Proposer une solution de rechange
 - Signaler un défaut à corriger ou une amélioration à apporter pour le compte du client.

<http://www.lgc.com/customersupport/supportservices/servicelevels/default.htm>

Centre de support

- Centre de service centralisé ou:
 - Tous les problèmes et les demandes de services sont acheminés
 - Ces problèmes et demandes sont réglés ou assignés
 - Ces problèmes et demandes de services sont suivis

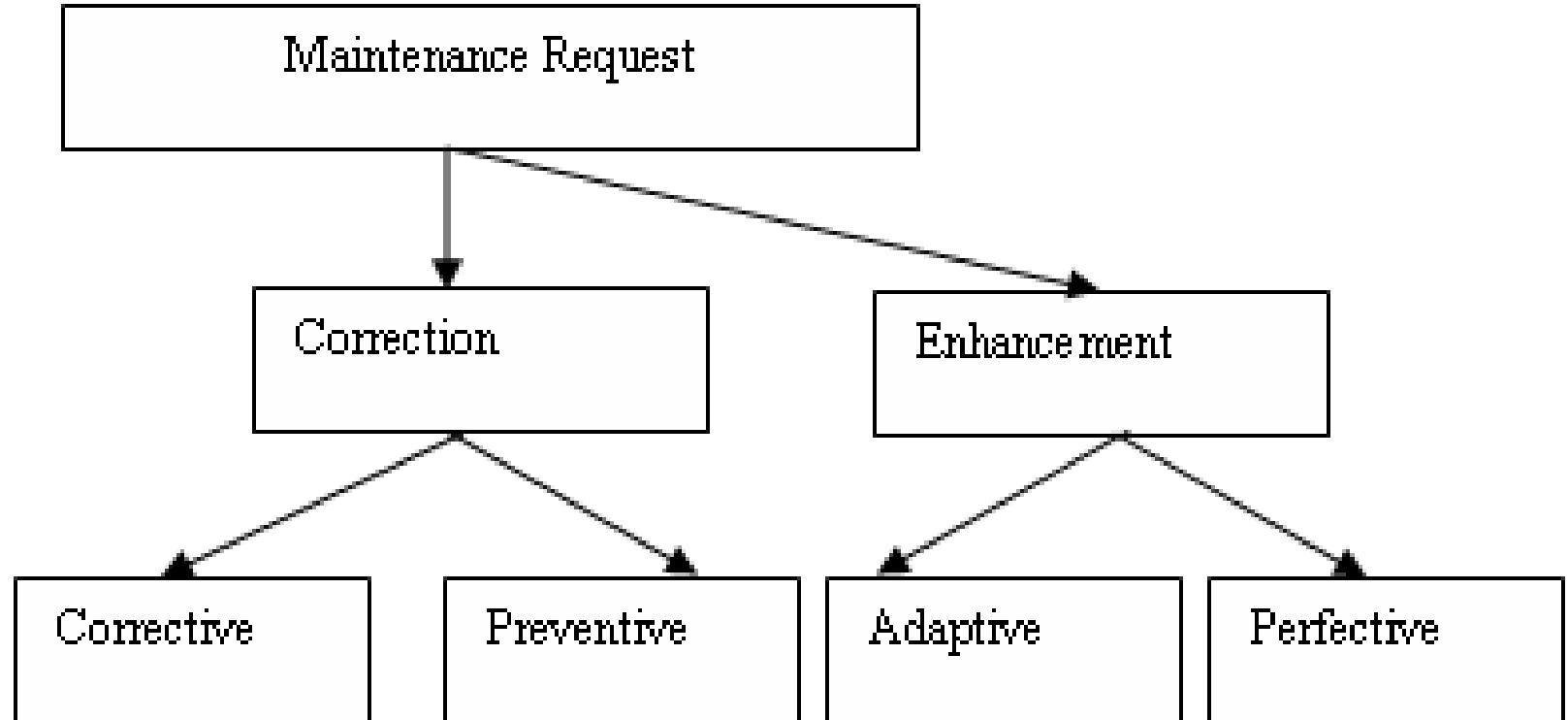
Processus

- Gestion des problèmes et des demandes
 - Communiquer
 - Documenter
 - Évaluer (en fonction des politiques)
 - Résoudre ou assigner
 - Suivre
 - Clore / Fermer

Niveau de service à l'interne

- Peut-être très utile pour prioriser la maintenance des systèmes patrimoniaux
- Peut aussi être utilisé pour mettre au rencart graduellement un vieux système.

Catégorie de requête

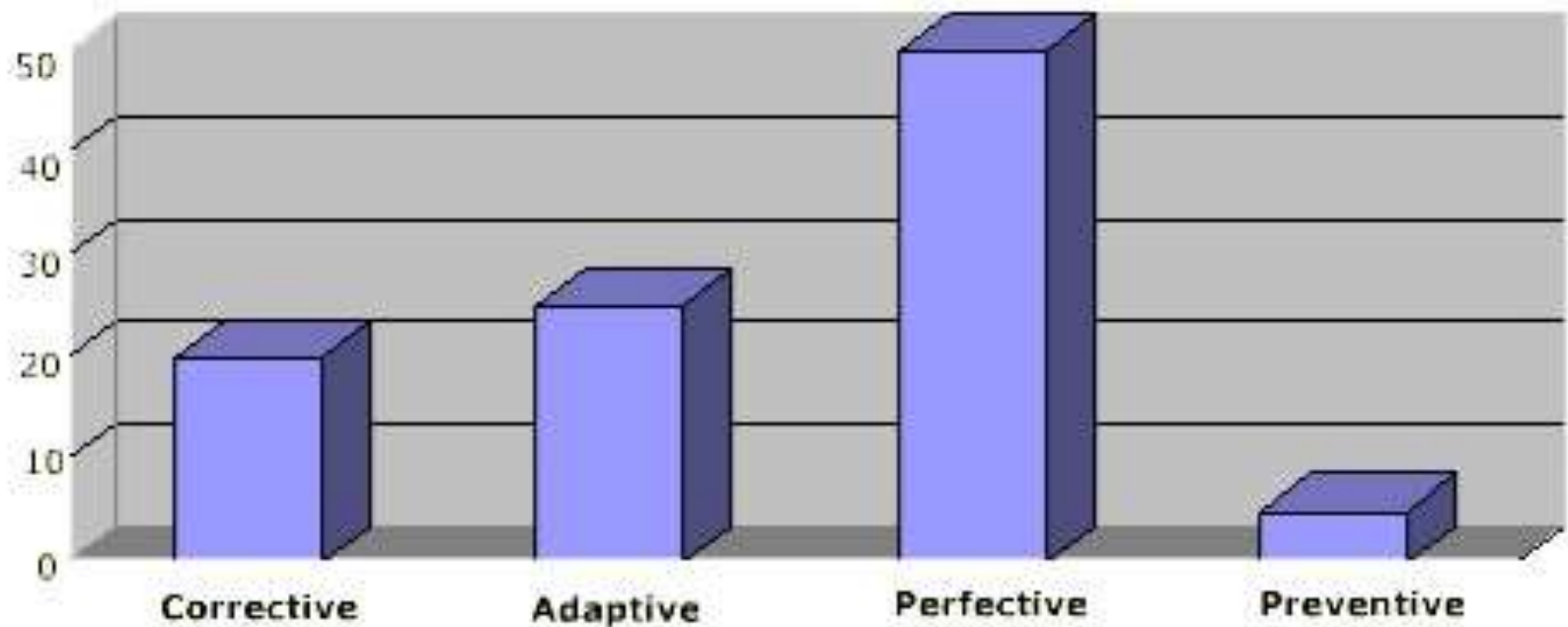


Types de maintenance

- **Corrective**
 - La correction d'une défectuosité découverte par la clientèle
- **Préventive**
 - Modifier un logiciel pour détecter et corriger des fautes avant qu'elles ne deviennent des défaillances
- **Perfective**
 - Améliorer les fonctionnalités d'un logiciel
- **Adaptative**
 - Modifier un logiciel afin de le mettre en relation avec d'autres applications

Types de maintenance

- Perfective: 50 %
- Adaptive: 25 %
- Corrective: 21 %
- Préventive: 4 %



Les défaillances ne sont pas la première source

- Les défaillances ne représentent pas le plus gros pourcentage de maintenance.
- Lorsque correctement mesuré:
 - 55 % requêtes sont des améliorations (Lientz)
 - Le plus gros pourcentage d'effort est dépensé dans l'ajout de fonctionnalité en réponse à un besoin d'affaires (Pressman)

Similitude avec logiciel

- Processus définis
- Il y a du code et des tests
- Il y a de l'assurance qualité
- Il y a de la gestion de la configuration

Activités distinctes

- Gestion du service et des événements
- Existence d'un SLA (parfois...)
- Transition
- Support opérationnel
- Résolution de problème
- Analyse d'impact
- Évolution et retrait

Caractéristiques de la maintenance

- Réception des requêtes aléatoires
 - Budget difficile à évaluer
- Charge de la maintenance
 - Gestion par files d'attente et non par gestion de projet
- Taille des requêtes limitées
- Assignment du travail dynamique
- Arrêt des travaux quand il y a une panne
- Contraintes face à une application déjà développée
- Attentes du client élevées

Différence avec le développement

- La situation est beaucoup moins favorable pour le mainteneur.
- Le système existe, avec ses défauts et ses contraintes architecturales. Il faut vivre avec.
- Les outils sont parfois désuets et ne supportent pas les techniques modernes de maintenance.
- Conséquemment, il y a moins d'options pour les améliorations.
- Les délais sont souvent courts.

Ressources

- S3M 
- April, A. et Abran, A. *Améliorer la maintenance du logiciel*. Loze-Dion éditeur, ISBN 292118088X, 2006, 337 p.

Questions?



TRAVAIL-2