

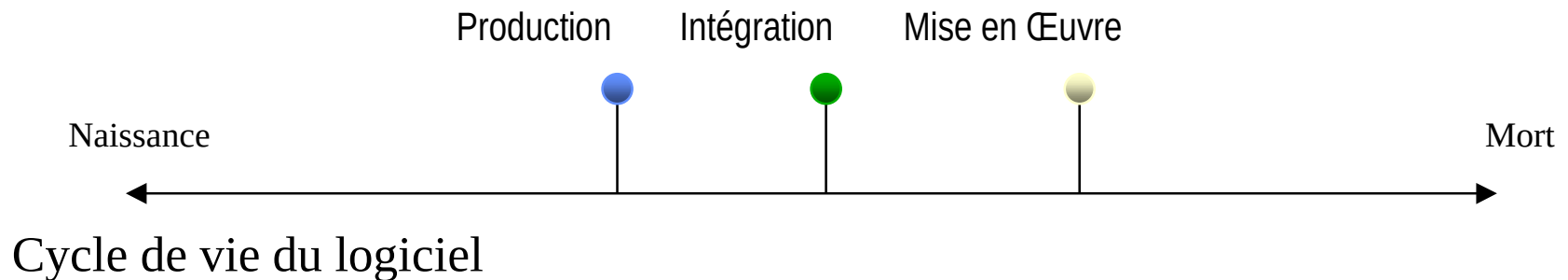
Analyse et conception des Systèmes d'Information

La démarche Merise : La Maintenance

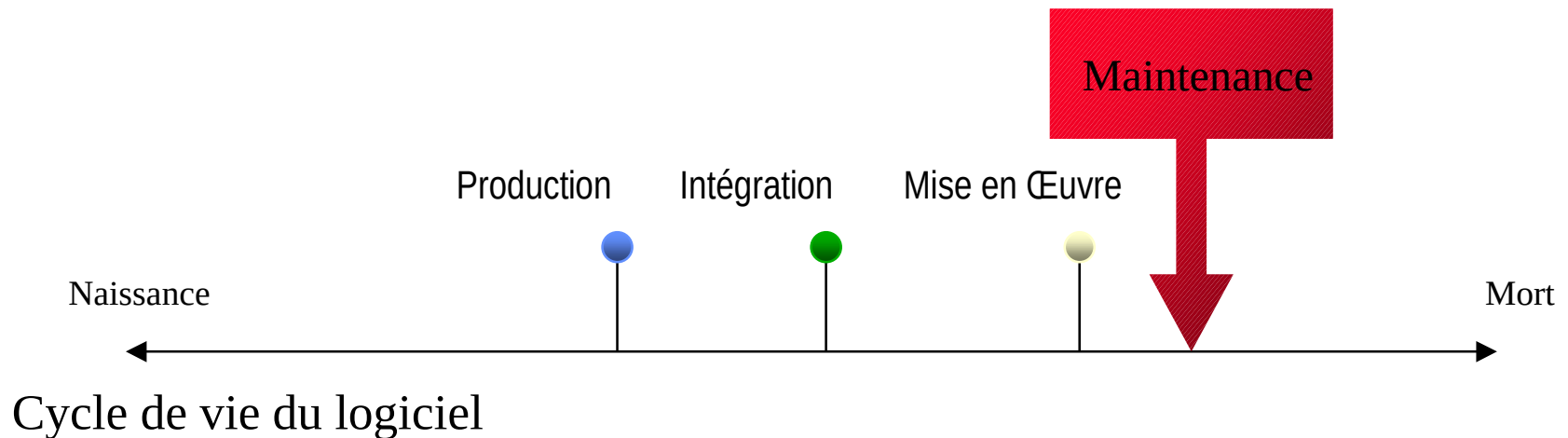
- **Place, spécificité, objectifs et principes directeurs**
- **Niveaux et catégories de maintenance**
- **Formes de maintenance**
- **Déroulement de l'étape de maintenance**
- **Organisation des équipes de maintenance**

- **Place**, spécificité, objectifs et principes directeurs
- **Niveaux et catégories de maintenance**
- **Formes de maintenance**
- **Déroulement de l'étape de maintenance**
- **Organisation des équipes de maintenance**

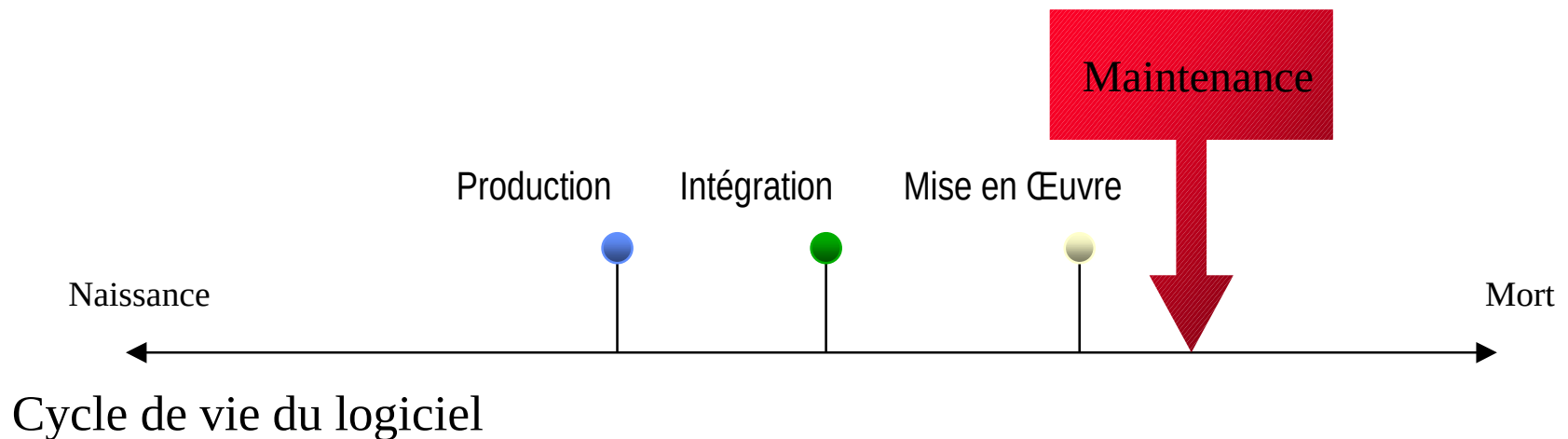
- En fin de cycle de vie du logiciel, après la production, l'intégration et la mise en œuvre



- En fin de cycle de vie du logiciel, après la production, l'intégration et la mise en œuvre

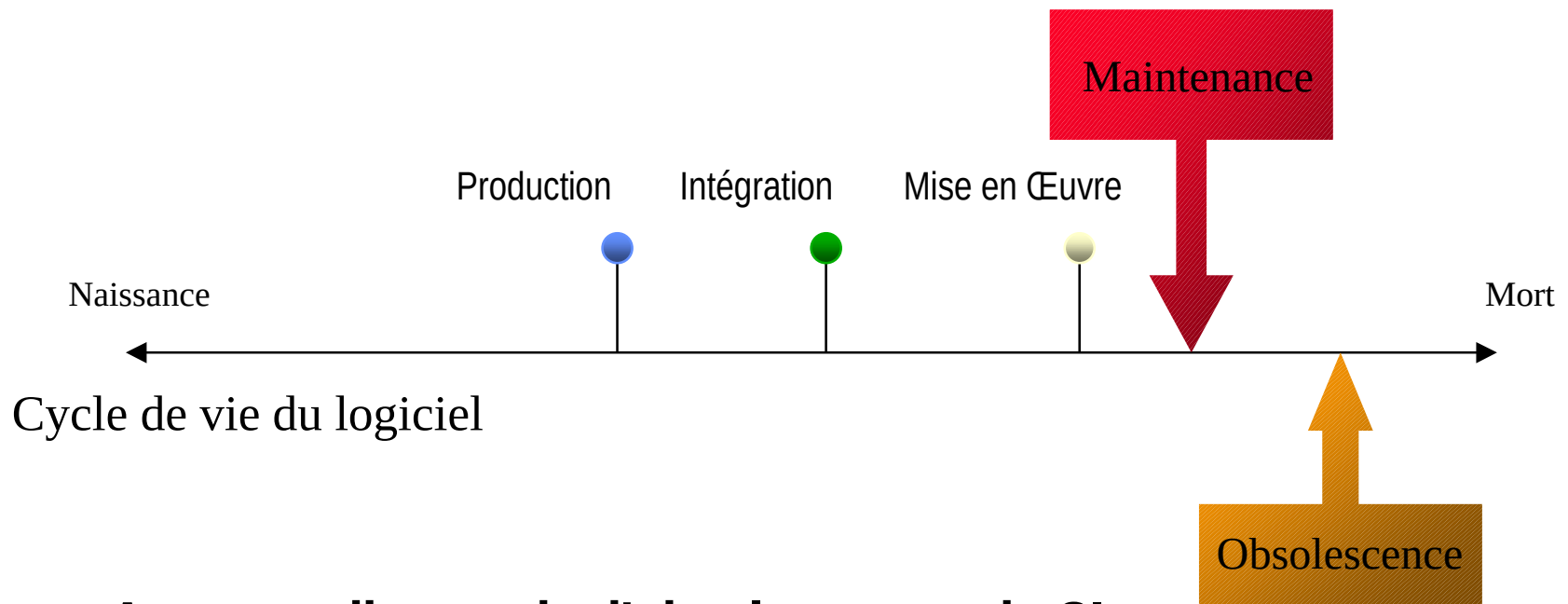


- **En fin de cycle de vie du logiciel, après la production, l'intégration et la mise en œuvre**



- **Avant un diagnostic d'obsolescence du SI**
 - (prévision du futur système d'information)

- En fin de cycle de vie du logiciel, après la production, l'intégration et la mise en œuvre



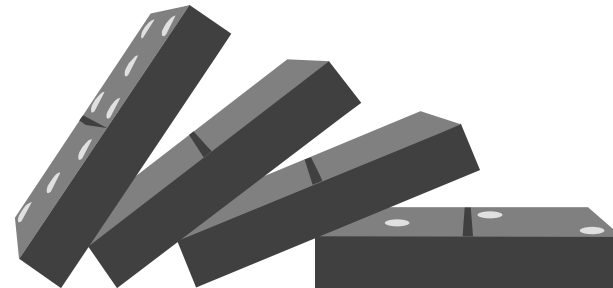
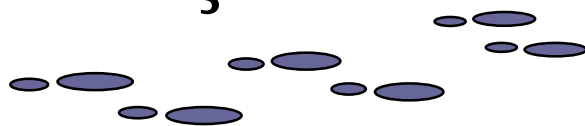
- Avant un diagnostic d'obsolescence du SI
 - (prévision du futur système d'information)

- Place, **spécificité**, objectifs et principes directeurs
- Niveaux et catégories de maintenance
- Formes de maintenance
- Déroulement de l'étape de maintenance
- Organisation des équipes de maintenance

- Trois aspects sont essentiels dans la maintenance :

 Visibilité du logiciel

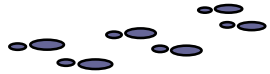
Traçabilité



Ergonomie de l'activité

- Mise en évidence des stratégies de développement du système d'information (schéma directeur)
- Etablissement d'un dossier de choix lors de l'étude préalable
- Définition d'une solution technique vue par l'utilisateur (étude détaillée)

La maintenance est définie selon Boehm comme l'ensemble des opérations de modification d'un logiciel opérationnel laissant intactes ses fonctionnalités de base.



- Complémentaire à la visibilité,
- Permet la remontée logique des programmes vers les éléments de plus haut niveau,
- Tout programme doit être amplement documenté.

- **Maintenance des nouvelles versions ou révision d'un logiciel en fonction des modifications de l'environnement**
 - Machine, système d'exploitation, ...



- **Pour chaque configuration doit être spécifié :**
 - les fonctionnalités prises en compte,
 - les besoins couverts des utilisateurs
 - les contraintes de développement sélectionnées
 - les produits logiciels la composant

- Place, spécificité, **objectifs** et principes directeurs
- Niveaux et catégories de maintenance
- Formes de maintenance
- Déroulement de l'étape de maintenance
- Organisation des équipes de maintenance

-  **Maintenir le système en fonctionnement**
-  **Rectifier les anomalies de fonctionnement et prendre en compte les demandes d'évolution**
-  **Etablir un scénario de développement des versions**
-  **Planifier les corrections apportées au système pour minimiser les coûts d'intervention**

-  **S'assurer du bon fonctionnement des nouvelles révisions avant tout nouvelle mise en exploitation**
-  **Procéder à cette mise en exploitation après l'accord des parties prenantes**
-  **Assurer une bonne gestion des différentes configurations du logiciels par la tenue à jour de la documentation du système**

- Place, spécificité, objectifs et principes directeurs
- Niveaux et catégories de maintenance
- Formes de maintenance
- Déroulement de l'étape de maintenance
- Organisation des équipes de maintenance

■ Suivre

pour détecter les
anomalies de
fonctionnement



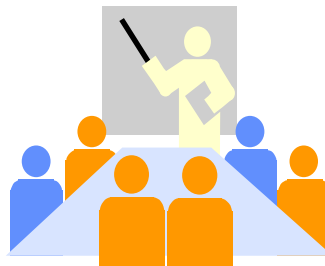
pour juger du caractère
opportun de la prise en
compte de nouveaux
besoins

■ Mener une politique de développement



- de versions et de révisions successives compte tenu d'un budget

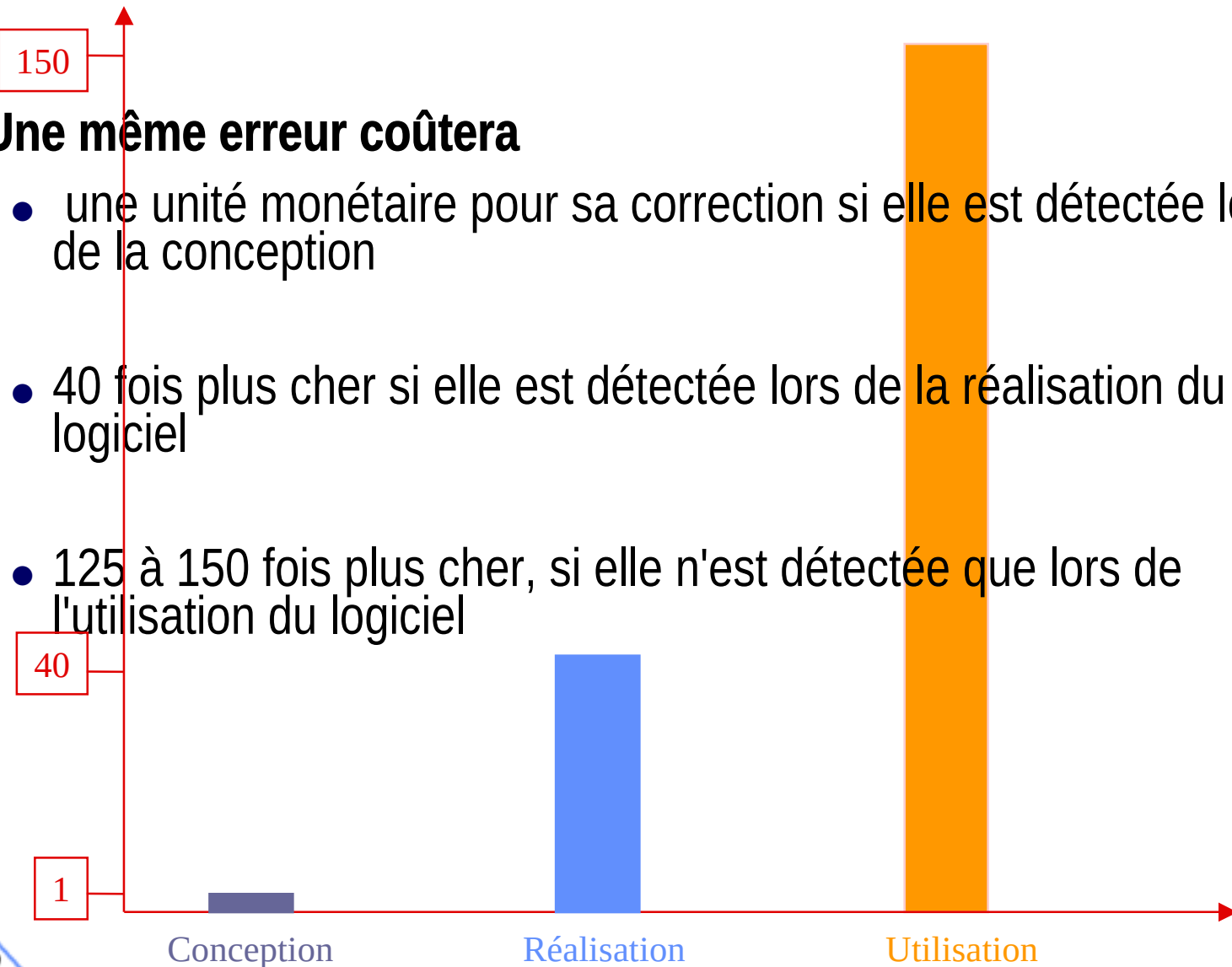
■ Réactualisation d'une réflexion de planification à moyen et long terme.



- **Conception difficile à comprendre, à modifier, à tester,**
- **Insuffisance de la documentation ou lacunes dans cette documentation,**
- **Inconsistance et non-correspondance entre la documentation et le code,**
 - voire entre les listings et le code exécutable
- **Absence de mémorisation des opérations de maintenance antérieure.**

■ Une même erreur coûtera

- une unité monétaire pour sa correction si elle est détectée lors de la conception
- 40 fois plus cher si elle est détectée lors de la réalisation du logiciel
- 125 à 150 fois plus cher, si elle n'est détectée que lors de l'utilisation du logiciel



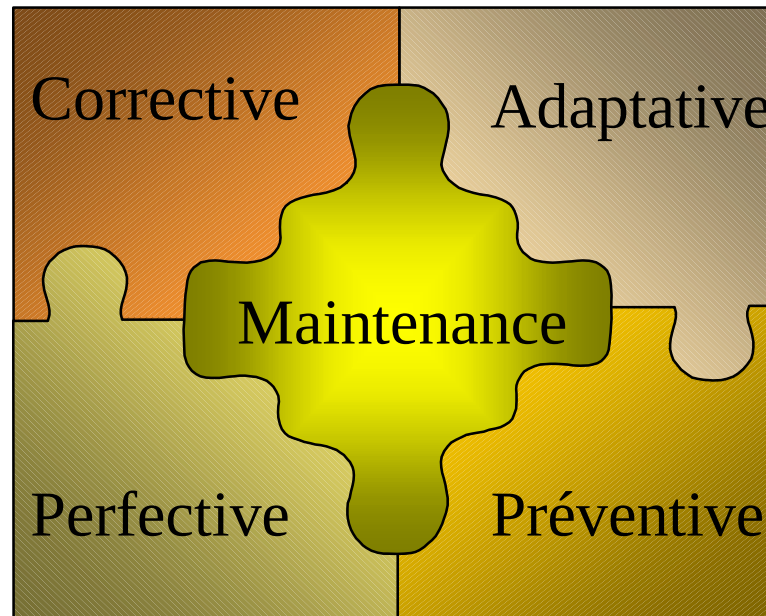
- **Estimation de la charge globale de maintenance (nombre de personne pour la maintenance)**

$$P = \frac{KISL}{RKMP}$$

KISL = Kilo Instructions Sources Livrées

RKMP = Ratio du nombre d'instructions sources maintenues par personne

- Place, spécificité, objectifs et principes directeurs
- Niveaux et catégories de maintenance
- Formes de maintenance
- Déroulement de l'étape de maintenance
- Organisation des équipes de maintenance



- **Elle ne porte que sur le logiciel développé**
 - Elle ne remet pas en cause les modèles de données ou de traitements
- **Corrections des erreurs du logiciels**
- **Représente de 17 à 20% du temps de maintenance**



■ Liée à l'environnement du logiciel

- contexte d'utilisation du logiciel,
- génération des ordinateurs,
- exploitation logiciel sur des matériels distincts
- ...

■ Evolution

- du modèle conceptuel des données
- du modèle conceptuel des traitements

■ Représente 18 à 25% du temps de maintenance



- **Catégorie de maintenance la plus fréquente (60%)**
- **Consiste à améliorer le fonctionnement du logiciel**
- **Entraîne une remise en question**
 - du modèle physique du logiciel
 - du modèle organisationnel des traitements



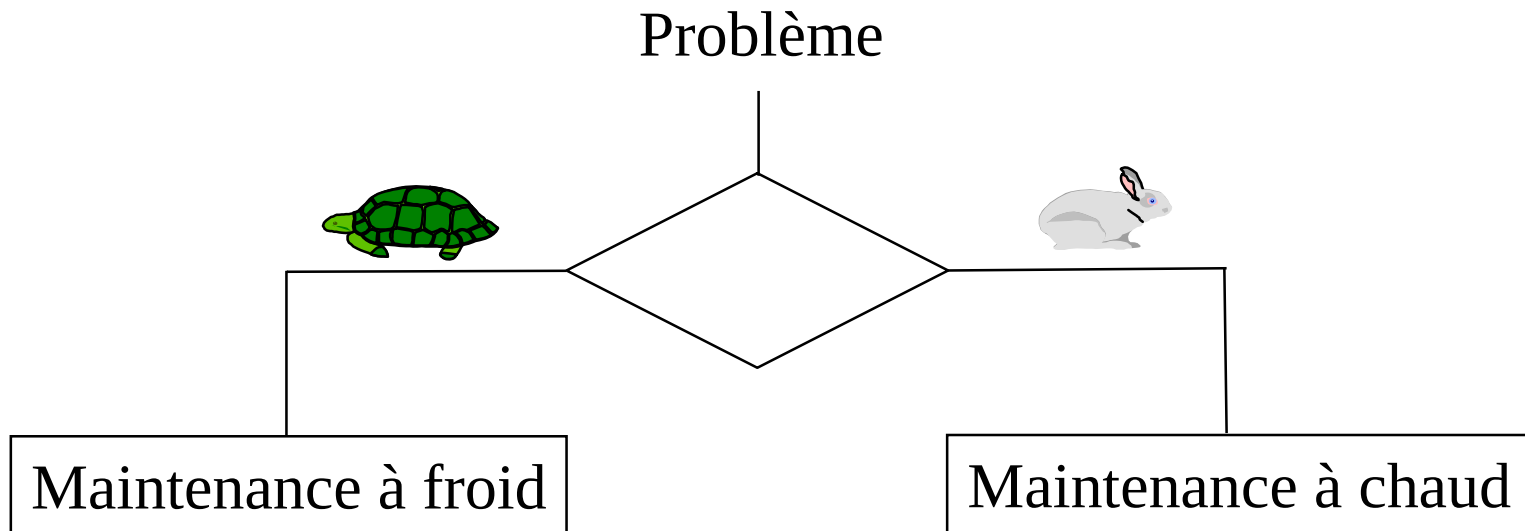
- **Diminuer le nombre des opérations de maintenance pour en diminuer le coût si :**
 - le système est développé dans un esprit de maintenabilité ultérieur,
 - le système est continuellement amélioré pour faire face et intégrer les nouvelles technologies
 - le système est maintenu en pensant à la maintenance ultérieure,.



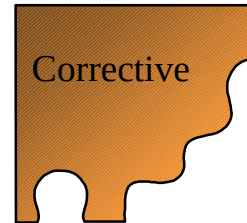
- Place, spécificité, objectifs et principes directeurs
- Niveaux et catégories de maintenance
- Formes de maintenance
- Déroulement de l'étape de maintenance
- Organisation des équipes de maintenance

■ Considération

- du temps
- de l'urgence des actions de maintenances



- **Suscitée par un fonctionnement défectueux du système d'information.**
- **Liée à**
 - la maintenance corrective
- **Exemple**
 - nouvelle version ou révision qui n'a pas pris en compte les caractéristiques particulières d'un équipement.



■ Liée à

- la maintenance adaptative
- la maintenance perfective

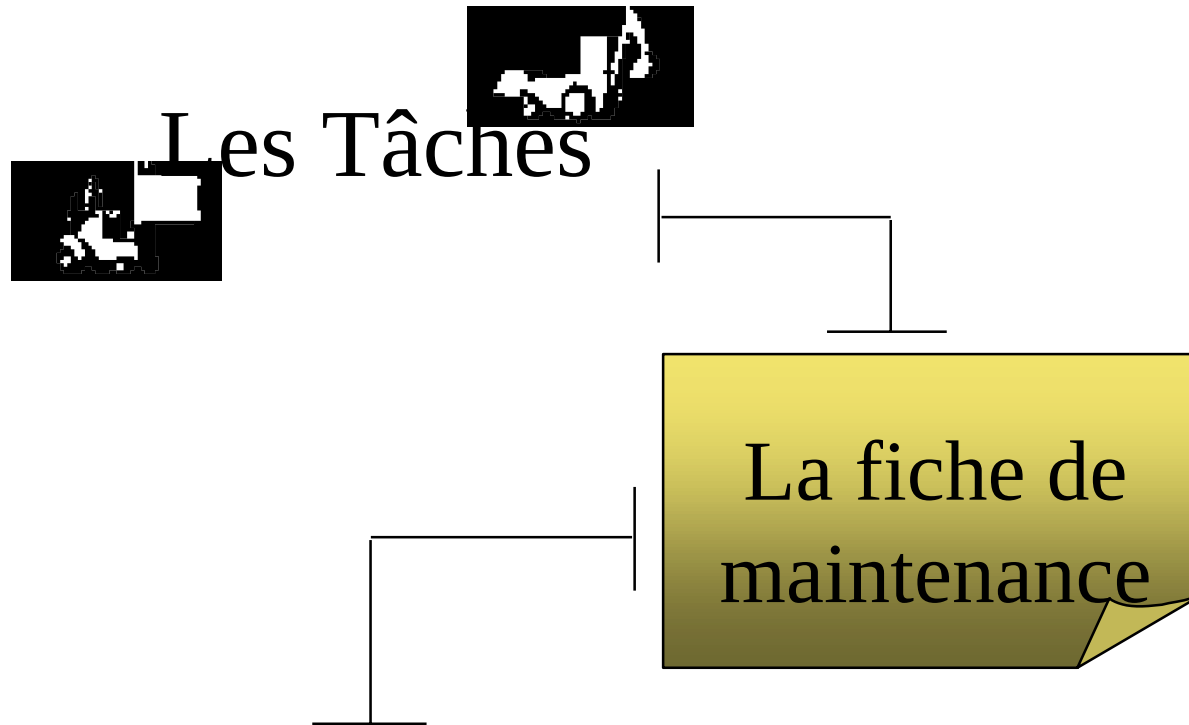


■ Caractéristiques

- un grand laps de temps sépare le diagnostic de la phase de réalisation des corrections/modifications.
- porte souvent sur la forme ou la présentation des résultats



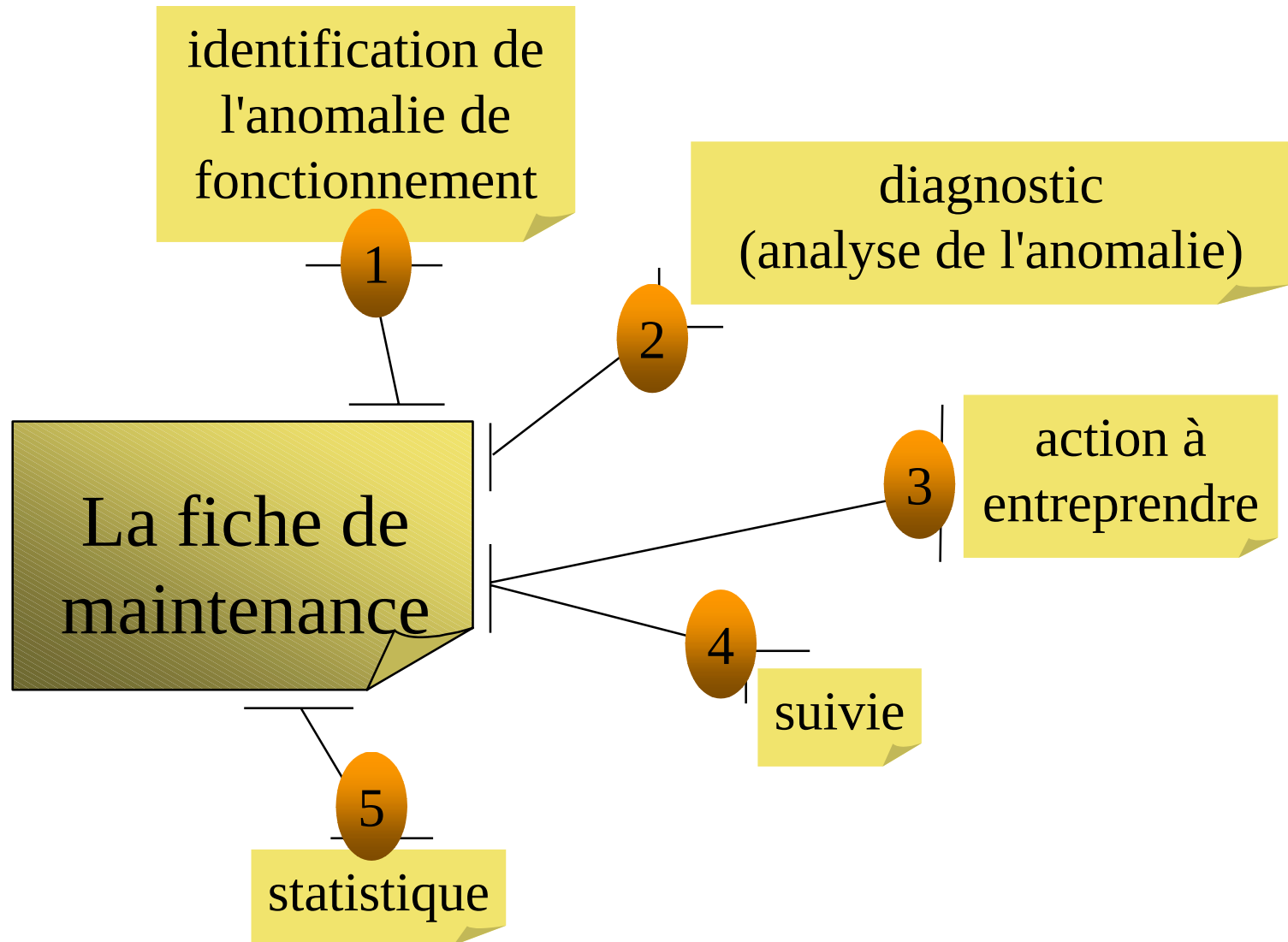
- **Place, spécificité, objectifs et principes directeurs**
- **Niveaux et catégories de maintenance**
- **Formes de maintenance**
- **Déroulement de l'étape de maintenance**
- **Organisation des équipes de maintenance**



Détail des
principales tâches

- **recueil de l'information,**
- **analyse des besoins,**
- **(re)conception du logiciel, voire du système,**
- **étude de l'impact de la modification,**
- **planning de la production du code,**
- **sollicitation de l'approbation des parties prenantes,**
- **production du code,**
- **test et intégration,**
- **(re)qualification du système.**





■ Une partie identification de l'anomalie de fonctionnement

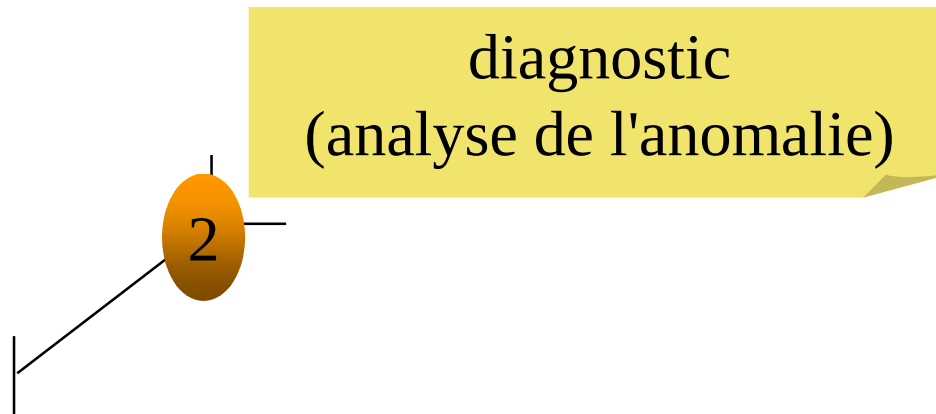
- identification de l'application
 - N° de version, N° de révision
- environnement de l'application
 - type ordinateur, configuration, description logiciel, ...
- conclut sur une estimation de l'importance et de l'urgence de la correction

identification de
l'anomalie de
fonctionnement

1

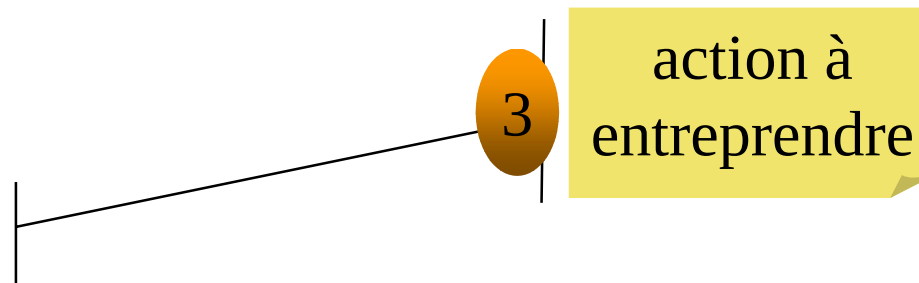
■ Une partie diagnostic (analyse de l'anomalie)

- nature de l'anomalie, programmes incriminés
- variables ou outils en causes, raisons, ...



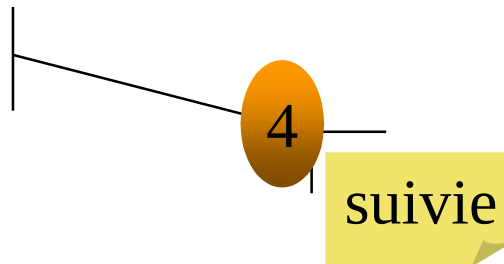
■ Une partie action à entreprendre

- exposé de l'action, identification des ressources prévues,
- coût estimé, moyens temporaires de contourner le problème
- date de prise en compte de l'anomalie, ...



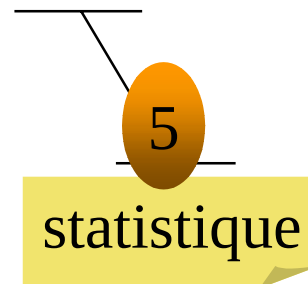
■ Une partie suivie

- identification ressources utilisées, corrections effectuées
- liste des éléments modifiés, coût après correction



■ Une partie statistique

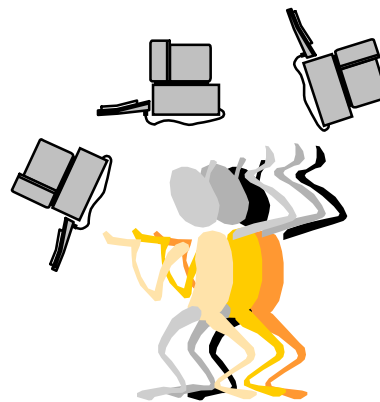
- liste de classes d'erreurs
- indice des erreurs les plus fréquentes
- indice de temps de diagnostic et de réparation par erreurs



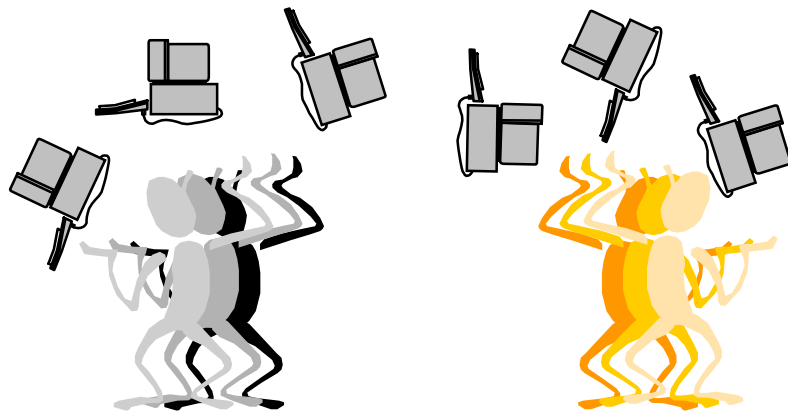
- **Recueil de l'information**
- **Analyse du besoin - diagnostic**
- **(re)Conception**
 - individus, relations modifiés
 - opérations, synchronisation, programmes retouchés
- **Etude d'impact**
 - identifie les actions à effectuer avant la mise en exploitation d'une nouvelle version
- **Planning de la production du code**

- **Approbation des parties prenantes**
 - accord interne du service de maintenance
 - accord des utilisateurs et des exploitants
- **Production du code**
- **Test locaux et intégration globale**
- **Approbation avant mise en œuvre**

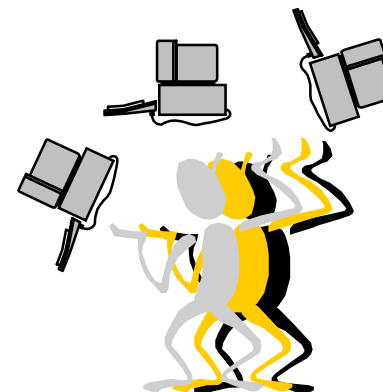
- **Place, spécificité, objectifs et principes directeurs**
- **Niveaux et catégories de maintenance**
- **Formes de maintenance**
- **Déroulement de l'étape de maintenance**
- **Organisation des équipes de maintenance**



Organisation intégrée



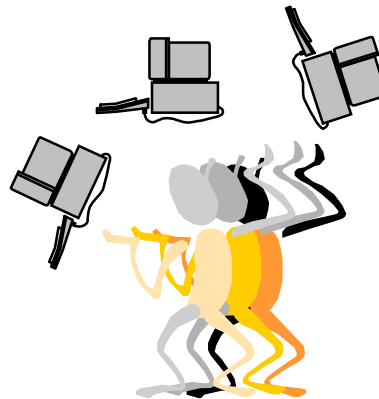
Organisation séparée



Mode mixte

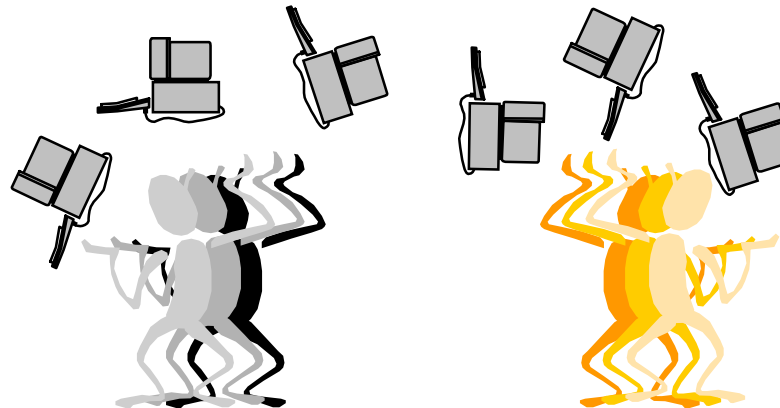
■ Organisation intégrée

- l'équipe de maintenance n'a pas d'existence propre en tant que telle
- Elle fait partie intégrante de l'équipe ou des équipes de développement



■ Organisation séparée

- l'équipe de maintenance à une existence propre et à peu ou pas de contact avec les équipes de développement



■ Mode mixte

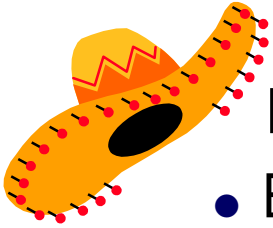
- un représentant de la maintenance est intégré dans les équipes de développement





Une expérience étendue

- Grandes connaissances des langages
- A la fois concepteur et programmeur, ingénieur système



La patience

- Ecoute de l'utilisateur, ...



La motivation personnelle



Le sens des responsabilités



L'humilité

- Capacité à faire face à toute critique (héros inconnu)



Le sens de l'innovation

- Donner une nouvelle forme au produit tout en minimisant l'impact des modifications



L'esprit "archiviste"

- garder trace des modifications successives
 - reconnaître la version courante d'un logiciel
 - reconstruire la version défailante d'un logiciel pour un utilisateur donné



L'esprit méthode

- travail avec méthode