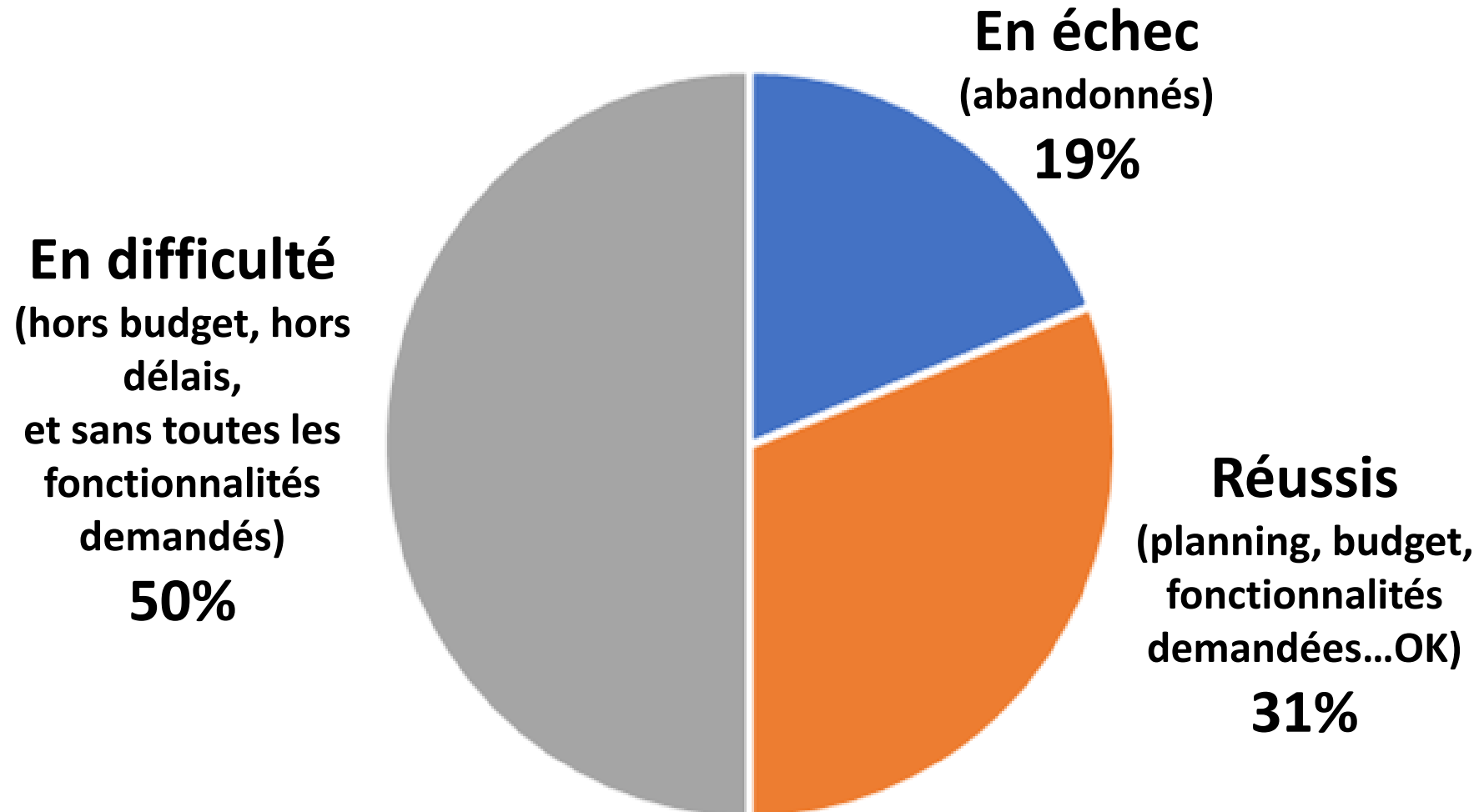




Déroulement d'un projet de système
d'information numérique (conventionnel) :
étapes, questions à se poser

Atelier SI2D
si2d.hypotheses.org

Taux de réussite des projets informatiques (Chaos report 2020 Standish Group)



Formes courantes de projets informatiques

- **Possibilité 1 : créer un nouvel logiciel**
 - Développé « en interne » ou par un prestataire externe (ESN)
- **Possibilité 2 : acheter un progiciel**
 - Choix d'acquérir un progiciel déjà existant sur le marché (ce progiciel devra être adapté aux spécificités de l'organisation)





Étapes d'un projet informatique

Les étapes d'un projet informatique

Cadrage



Planification / Conception



Réalisation



Livraison et recette



Déploiement



Maintenance/évolution

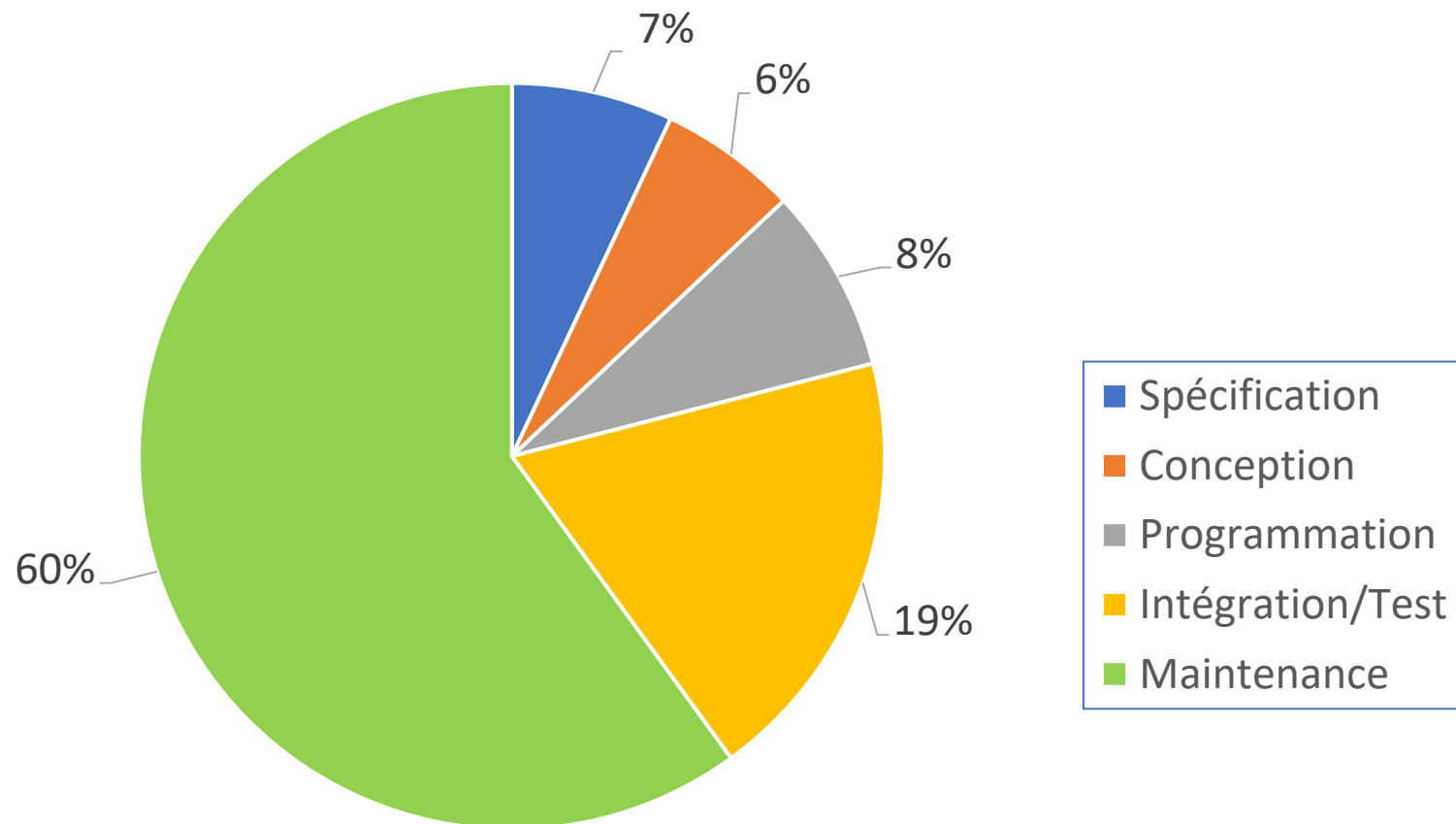
- Etude de faisabilité et d'opportunité
- Définition des objectifs et du périmètre du projet
→ *Dossier de cadrage*
- Organisation du projet (équipe, comités)
- Planification et budget prévisionnel
→ *Cahier des charges*
- Achat d'un progiciel : cycle spécifique
- Logiciel spécifique : cycle de développement logiciel
→ *Logiciel ou progiciel et documentation*
- Tests techniques
- Tests fonctionnels
→ *Validation de la réception*
- Communication
- Formation / Accompagnement au changement
→ *Mise en service du logiciel ou progiciel*
- Maintenance technique
- Gestion des évolutions fonctionnelles et d'IHM
→ *Jusqu'à la fin de vie du système*

Cycle de vie d'un système

- Trop souvent, la gestion d'un projet informatique s'arrête au déploiement du système
- Or, c'est le **cycle de vie** du système qui doit faire l'objet d'une gestion structurée
 - des évolutions seront nécessaires tout au long de la vie du système
 - Cf. Pierre Caye* qui qualifie la maintenance de « Développement évolutif continu »

* *Durer*, Ed. les Belles Lettres, 2020

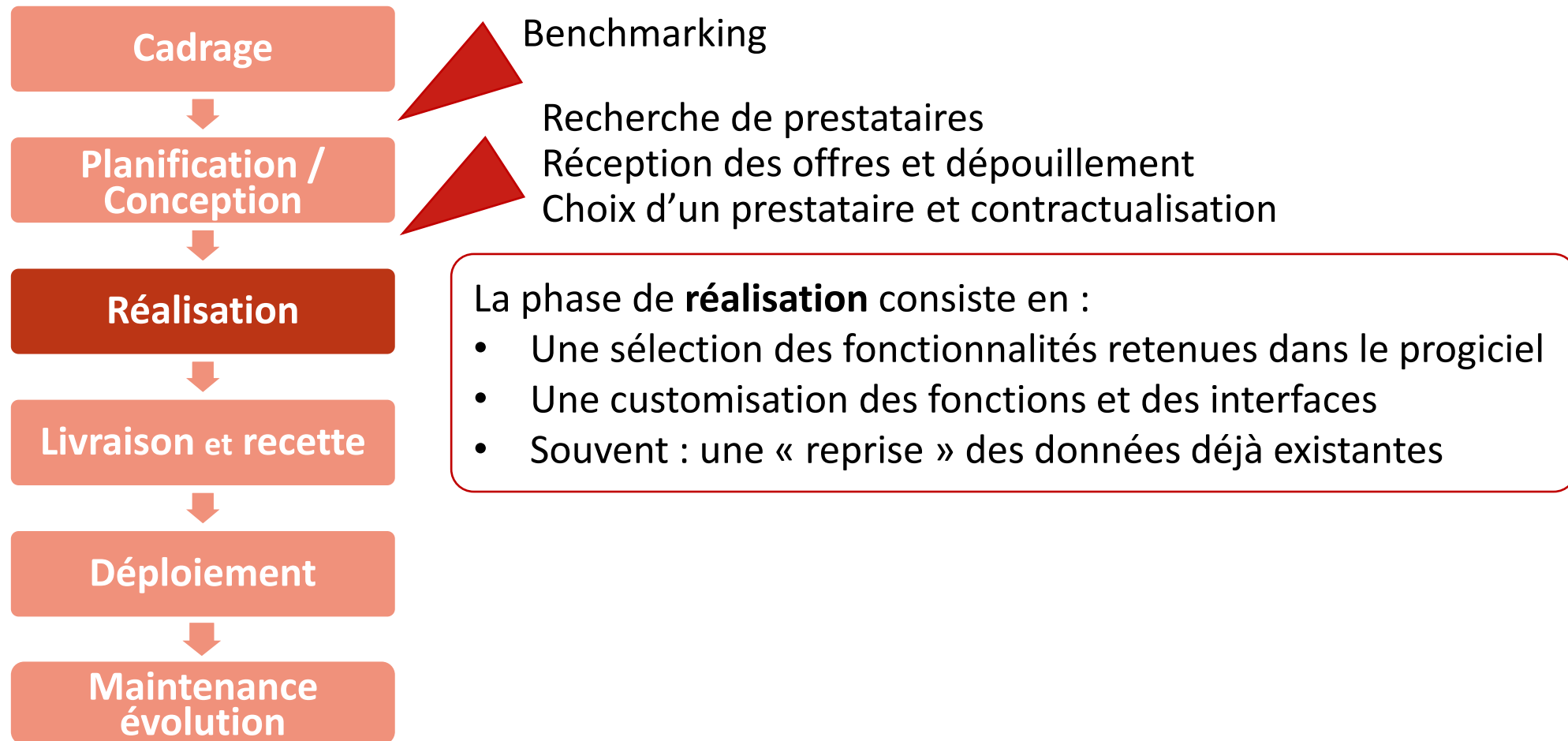
Projets informatiques : répartition de l'effort



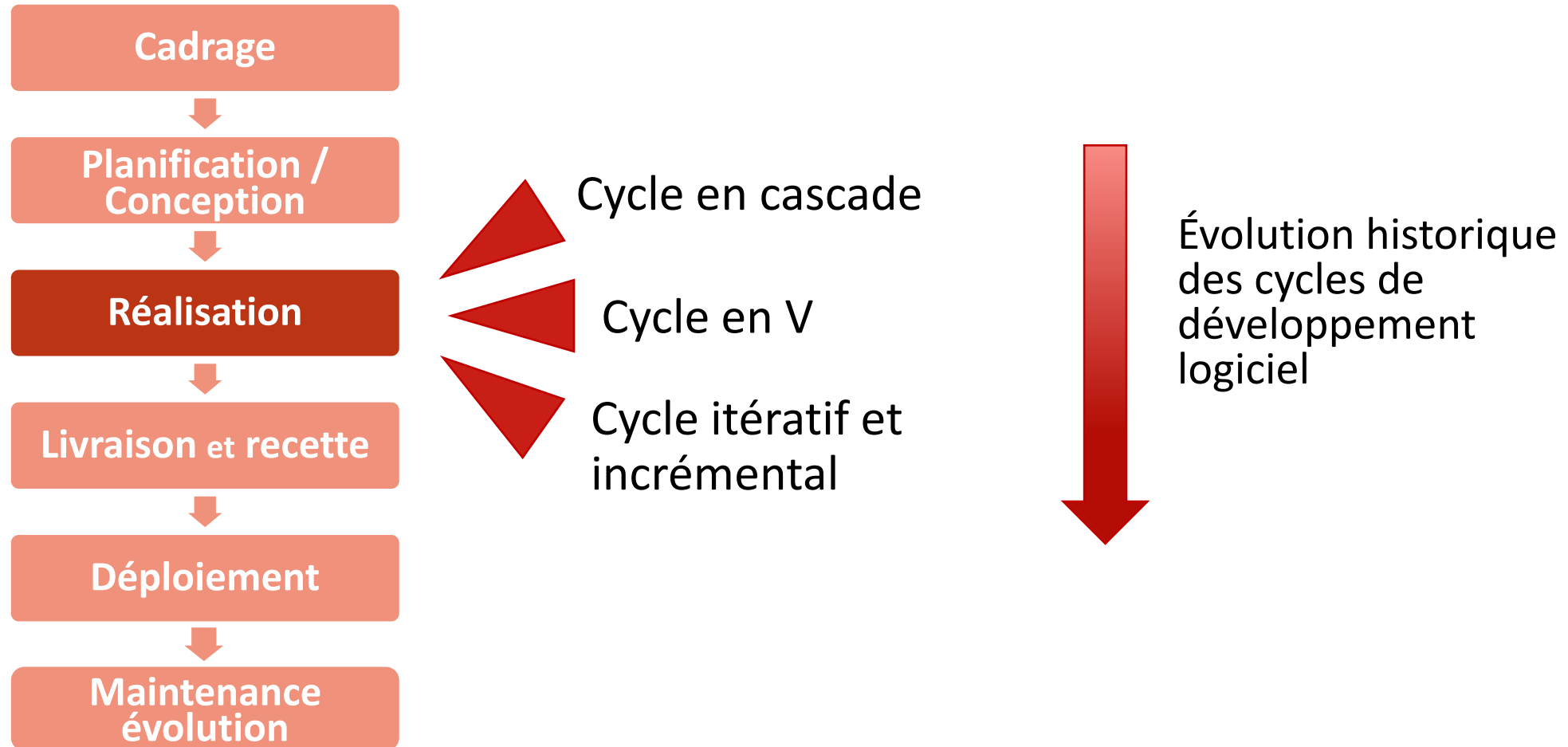


Cycles de développement logiciel

Cycle spécifique pour l'achat du progiciel



Cycles de développement logiciel



Les trois principaux types de cycles de développement logiciel

- Cycle de développement : phases nécessaires à la réalisation du logiciel
- Les trois principaux types de cycles :
 - Le **cycle en cascade** :
 - principe de division, enchaînement séquentiel, une étape ne peut remettre en cause que la précédente
 - Le **cycle en V** : une attention particulière à la validation
 - enchaînement séquentiel, mais les étapes amont doivent intégrer la préparation des étapes en aval pour vérifier la conformité du logiciel aux besoins
 - Le **cycle itératif et incrémental (en spirale)** : une limitation de « l'effet tunnel »
 - après une première analyse des besoins, les développements sont menés avant que l'intégralité des spécifications n'aient été faites, livraisons fréquentes avant la fin du développement pour limiter l'effet tunnel
- Les étapes communes :



Les étapes du cycle

Analyse

- Analyse de l'existant et analyse des besoins
 - Que doit faire le logiciel ? Quelles sont les contraintes ?
- *Cette étape s'appuie sur le cahier des charges, il en résulte un dossier d'analyse*

Conception

- Conception architecturale et conception détaillée
 - Comment le logiciel est construit ? Quels algorithmes ? Quelle structure de données
- *Résultat : le dossier de conception*

Développement

- Codage : construction d'un ensemble de programmes
 - On effectue des tests unitaires à cette étape : est-ce que l'algorithme fonctionne ?
- *Résultat : le logiciel développé*

Test

- Différents tests : tests d'intégration, recette des utilisateurs
 - On vérifie que l'ensemble des programmes fonctionnent ensemble
 - On vérifie que le logiciel correspond aux besoins des utilisateurs
- *Cahier de recette*

Déploiement

- Installation du logiciel sur les postes ou ouverture de l'accès aux logiciels
- Livraison de la documentation



Acteurs et instances

Les acteurs des projets informatiques

- Côté besoin (fonctionnel)
 - Le commanditaire : celui qui prend la décision de faire développer ou d'acheter un logiciel
 - Le client : Celui qui paye pour le produit
 - Les utilisateurs : les futurs utilisateurs du logiciel (utilisateurs directs ou indirects)
 - Le chef de projet fonctionnel (MOA)
- Côté solution
 - En cas d'achat de progiciel : une société éditrice ou intégratrice de progiciels
 - Commerciaux, consultants métier, consultants techniques
 - En cas de développement de logiciel :
 - Architectes, développeurs, concepteurs et chef de projet technique (en interne ou chez un prestataire)
- En accompagnement : une assistance à maîtrise d'ouvrage (AMO)

Les instances des projets informatiques

1/2

- L'instance décisionnaire : comité de pilotage
 - Le commanditaire
 - Le client ou son représentant
 - Le chef de projet fonctionnel
- Précise les objectifs du projet
- Définit la stratégie à suivre pour le projet (ex : achat d'un progiciel ou développement spécifique)
- Fait les arbitrages (ex : planification du projet, acteurs impliqués)
- Valide la/les livraisons

Les instances des projets informatiques

2/2

- Le suivi opérationnel : comité de suivi ou comité de projet
 - Le chef de projet fonctionnel
 - Le chef de projet technique
 - Le commanditaire ou son représentant
 - D'autres parties prenantes peuvent être intégrées : comité d'utilisateurs, membres de l'équipe technique, etc.
 - suivi de l'avancement du projet, coordination
- Le comité d'utilisateurs
 - Échantillon de futurs utilisateurs impliqués de manière plus ou moins importante dans les différentes étapes du projet
 - Comment sont-ils choisis ? Par leurs pairs, par la direction... ?
 - Son rôle est à minima d'effectuer la recette

Enjeux de démocratie et d'efficacité : questions 1/2

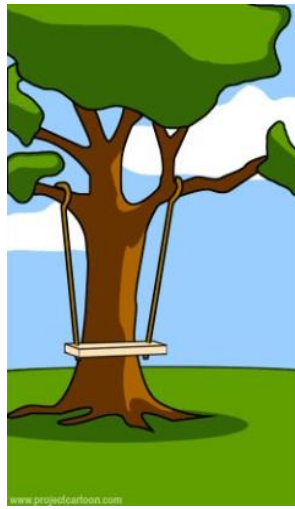
- Cadrage
 - Pourquoi lance-t-on ce projet informatique ?
 - Quels sont les objectifs ? Sont-ils explicites ?
 - Qui les définit ? Avec quelle participation des utilisateurs, quelles possibilités d'amendements ou de remise en cause ?
 - Qui est informé des objectifs ?
- Planification
 - Quel est le rôle de chaque comité ?
 - Qui en fait partie ? Qui décide de la composition ? Qui sont les décisionnaires ? Quelle information transmise et qui en décide ?
 - Quelle stratégie pour la réalisation ?
 - Développement d'un logiciel spécifique (en interne ou avec un prestataire ?) ? Si prestataire : quels critères de choix ?
 - Achat d'un progiciel ? Quels critères de choix ?
 - Qui rédige le cahier des charges ?

Enjeux de démocratie et d'efficacité : questions 2/2

- Réalisation
 - A chaque étape du cycle ses enjeux : Quels modèles ? Quels choix technologiques ?
 - Pour quel cycle de développement opter ?
 - Si choix d'un cycle de développement itératif et incrémental (agile)
 - Qui est client (Product Owner) ?
 - Quelle implication pour les utilisateurs ?
- Tests et déploiement
 - Quelle stratégie de déploiement dans l'organisation ? Quelle communication ?
 - Quel accompagnement au changement ? Quelles formations et pour qui ?
 - Quelle gestion des retours des utilisateurs lors de cette phase ?
- Maintenance et évolution
 - Quelle gestion de projet prévue pour les évolutions ?
 - Quelle(s) instance(s) pour cette gestion ?
Ex. : le comité d'utilisateurs est-il maintenu ?
 - Quelle(s) procédure(s) pour faire remonter les demandes ? Quelle priorisation ? Qui décide du traitement des demandes ?
 - Possibilité d'un retour sur les objectifs initiaux du système ?



Comment le client a expliqué son besoin



Comment le chef de projet l'a compris



Comment l'analyste l'a conçu



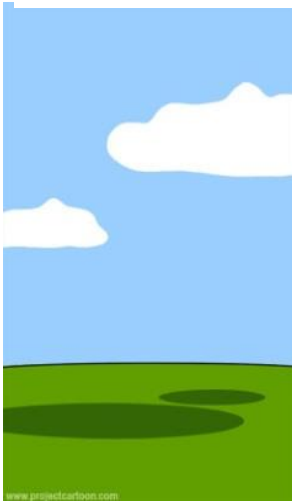
Comment le développeur l'a programmé



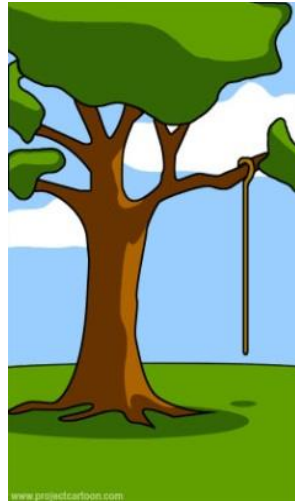
Comment le responsable des tests l'a reçu



Comment le commercial l'a décrit



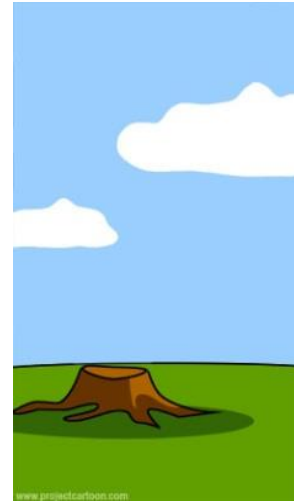
Comment le projet a été documenté



Quelles fonctionnalités ont été installées



Comment le client a été facturé



Comment le système a été maintenu



Comment le marketing en a fait la publicité



Ce dont le client avait réellement besoin