

Méthodes Agile

Les fondamentaux

Fondamentaux et pratiques

Plan

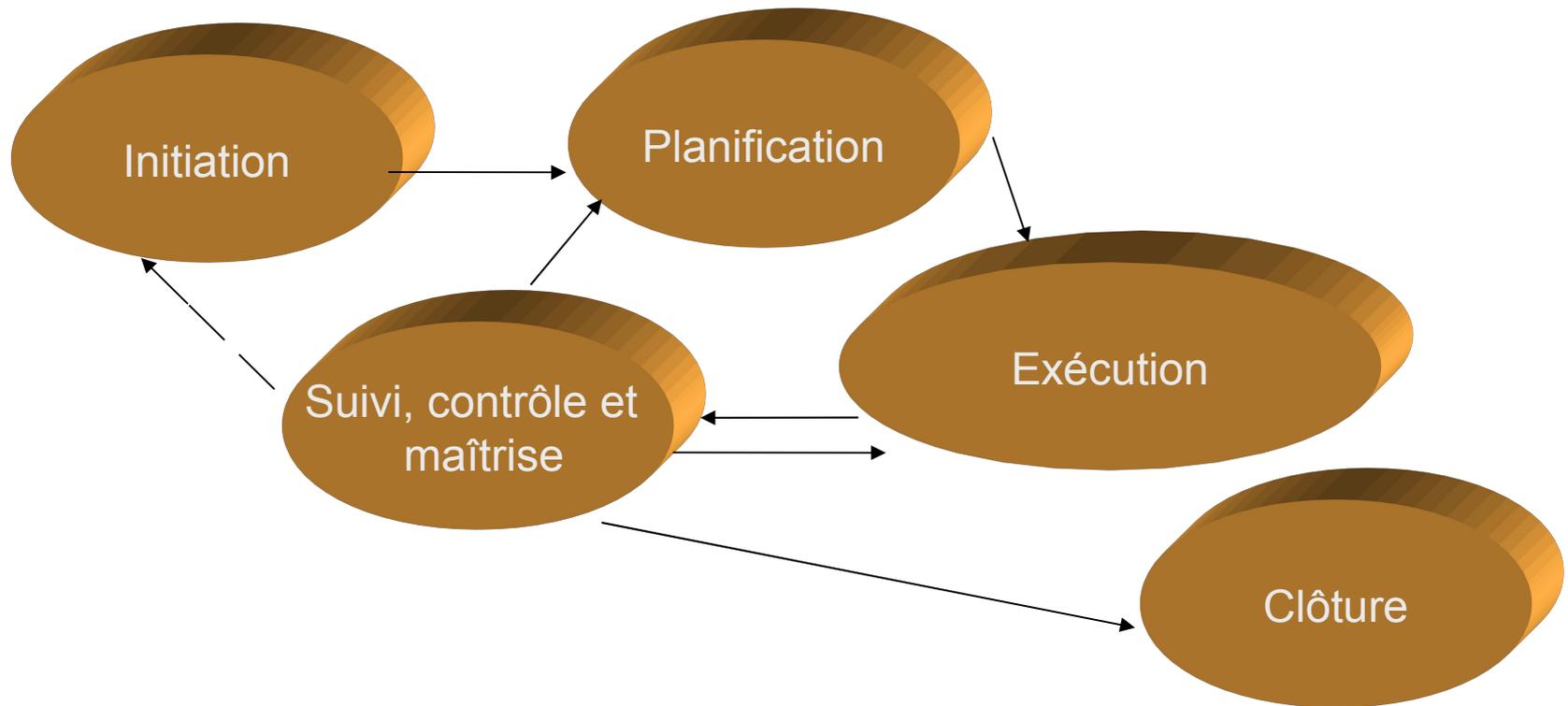
- ❖ **Introduction et révision des concepts de base**
- ❖ **Modèle de gouvernance**
- ❖ **Cycle en cascade**
- ❖ **Méthodes Agile**
- ❖ **Principales méthodes Agile**
- ❖ **Avantages**
- ❖ **Synthèse**
- ❖ **Différences entre méthodes traditionnelles et méthodes Agile**

Introduction et révision des concepts de base

- **Le management de projet est effectué en appliquant les 42 processus de la méthodologie du PMI groupés logiquement dans les 5 groupes de processus**

Introduction et révision des concepts de base

Les 5 groupes de processus



Introduction et révision des concepts de base

❖ On identifie 9 disciplines ou champs de connaissance (knowledge areas)

- L'intégration (integration)
- Le contenu (scope)
- Les délais (time)
- Les coûts (cost)
- La qualité (quality)
- Les ressources humaines (human resources)
- La communication (communication)
- Les risques (risk)
- Les approvisionnements (procurements)

Introduction et révision des concepts de base

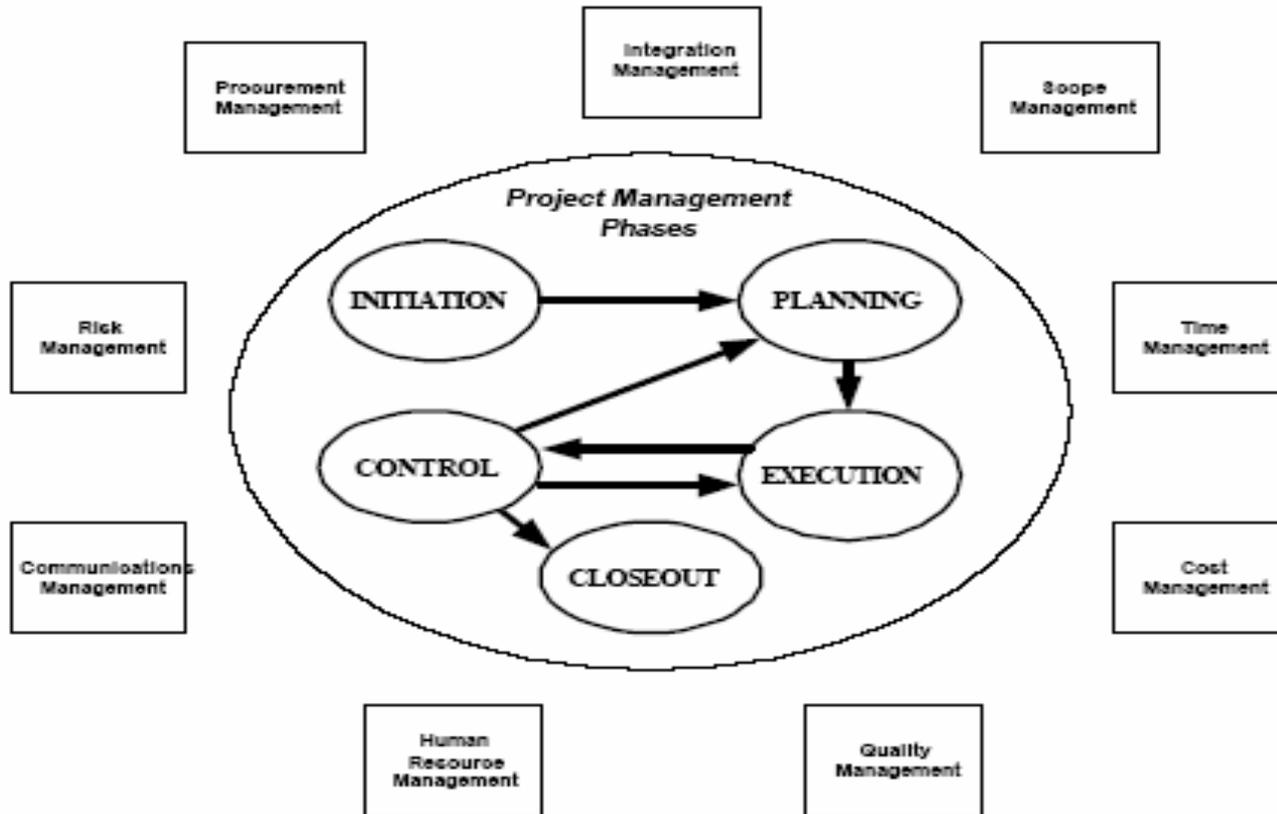
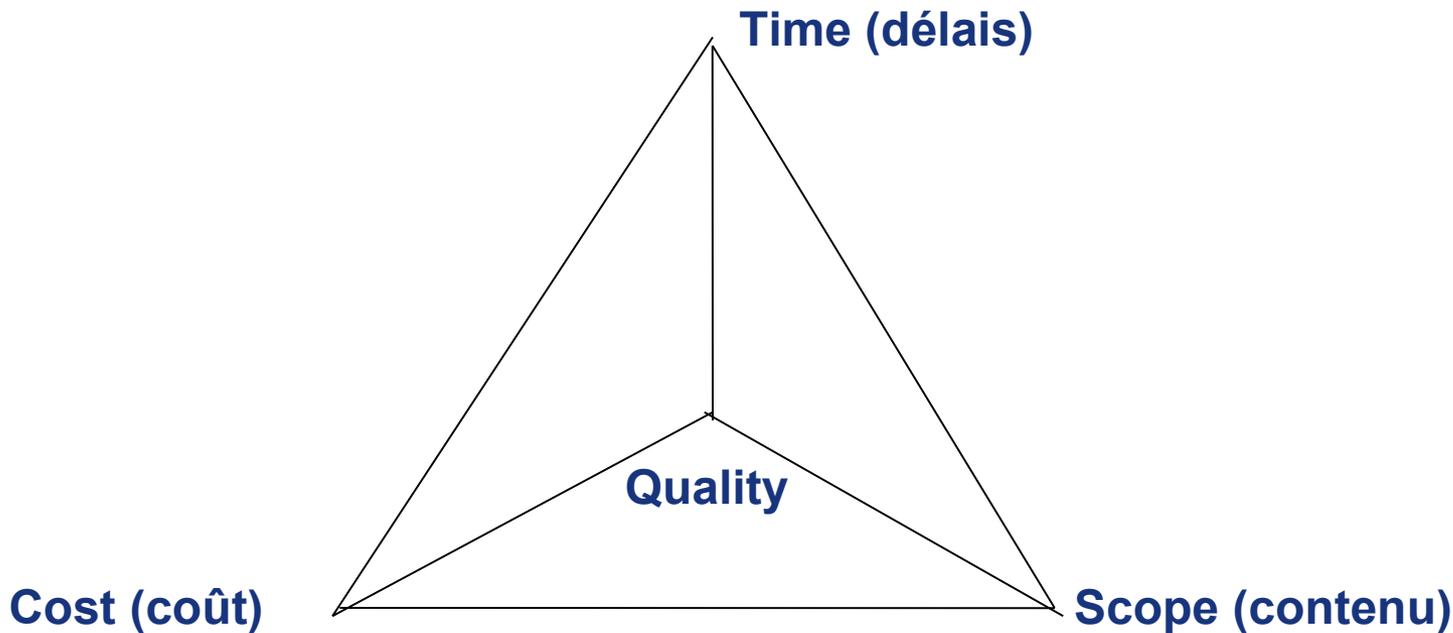


Figure 1.1
State of Michigan Project Management Knowledge Areas and Phases

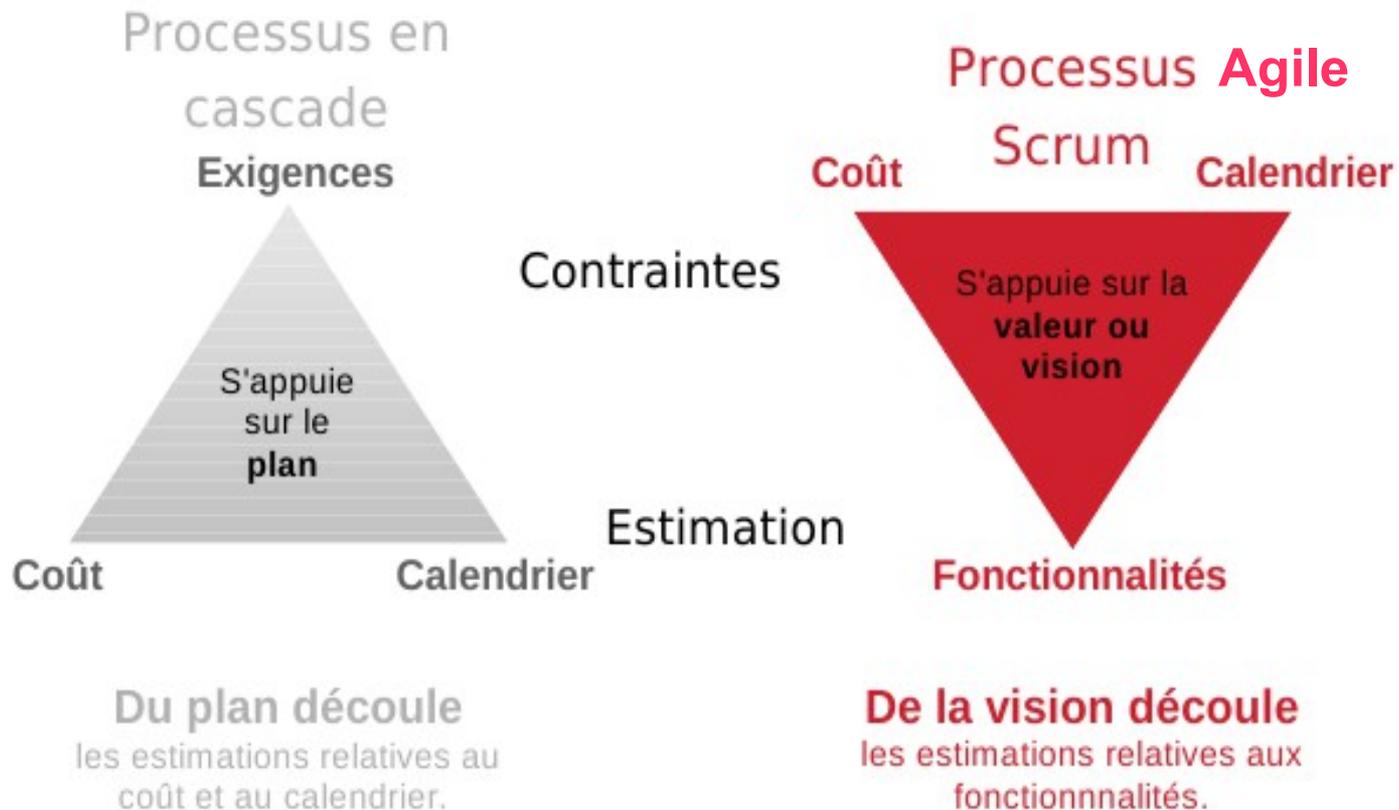
Introduction et révision des concepts de base

❖ La triple contrainte



Modèle de gouvernance

❖ Modèle de gouvernance



Modèle de gouvernance

❖ Modèle de gouvernance

Predictive

Start with Plan
and all
requirements



End with all
requirements
completed

Scrum - Empirical

Start with Goals
and some
priority
requirements



End with
Goals met

Le cycle en cascade

- ❖ **Le cycle en cascade se caractérise par des phases séquentielles, qui se succèdent après la validation des livrables produits par la phase précédente:**
 - **Recueil des besoins**
 - **Analyse détaillée**
 - **Conception détaillée**
 - **Développement**
 - **Tests, contrôle qualité**
 - **Intégration**

Le cycle en cascade

- ❖ **Les failles du cycle en cascade:**
 - **Rigidité de l'approche**
 - **Effet tunnel**
 - **Mauvaise communication**
 - **Levée tardive des facteurs à risques**
 - **Documentation pléthorique**

Difficultés des méthodes prédictives

▸ Instabilité des besoins

- Entre 20% et 40% des besoins évoluent au cours du projet

▸ Complétude des spécifications dès le début du projet

- 56% des anomalies sont liés à la compréhension des besoins

▸ Imprécision des macro-chiffrages

- Jugement d'expert contractualisé

▸ Pilotage par les plans

- Pas de changements possibles
- Pas de places aux impondérables
- Planning non réaliste

▸ Effet tunnel / Boite noire

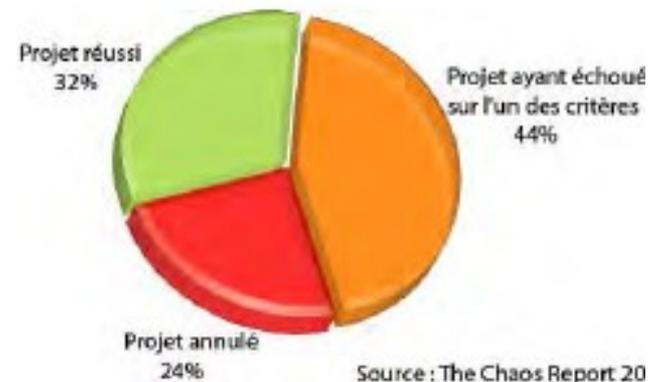
- Les résultats et les tests arrivent très tard
- Peu de possibilité de pilotage pour le client

▸ Qualité

▸ Implication et motivation des acteurs

▸ Monde moderne

- Typologies projets, TTM, livraisons fréquentes, concurrence, nouveau client...



Les méthodes Agile

- ❖ **Le pourquoi des méthodes Agile:**
 - Réduire le temps de mise sur le marché (Time to market)
 - Améliorer la qualité
 - Réduire les activités sans réelle valeur ajoutée
 - Travailler sur des éléments de plus grande valeur
 - Avoir une meilleure prévisibilité
 - Avoir du plaisir au travail

Les méthodes Agile

- ❖ **Caractéristiques des méthodes Agile:**
 - Une méthode Agile est une approche itérative et incrémentale, qui est menée dans un esprit collaboratif, avec juste ce qu'il faut de formalisme
 - Elle génère un produit de haute qualité tout en prenant en compte l'évolution des besoins du client

Les méthodes Agile

- ❖ **Origine et valeurs des méthodes Agile:**
 - **Mouvement né en 2001 aux USA**
 - **Réunion de 17 experts en développement logiciel**
 - **But: trouver un socle commun de valeurs et de bonnes pratiques**
 - **Résultats: écriture du Manifeste pour le développement logiciel Agile et création de l'Agile Alliance**
 - **association chargée de la promotion de l'agilité et du soutien aux équipes**

Les méthodes Agile

❖ Le manifeste Agile:

- Nous découvrons de meilleures approches pour le développement informatique en en faisant nous même, ainsi qu'en aidant les autres dans leur travail

Les méthodes Agile

❖ Le manifeste Agile:

- A travers ce travail, nous portons plus de valeurs sur
 - “les individus et leurs interactions” avant “les processus et les outils”
 - “des fonctionnalités opérationnelles” avant “la documentation »
 - « collaboration avec le client » plutôt que « contractualisation des relations »
 - « acceptation du changement » plutôt que « conformité au plan »

Les méthodes Agile

❖ Le manifeste Agile:

- Les valeurs mises en avant à gauche n'excluent pas les valeurs à droite
 - “les individus et leurs interactions” avant “les processus et les outils”
 - “des fonctionnalités opérationnelles” avant “la documentation »
 - « collaboration avec le client » plutôt que « contractualisation des relations »
 - « acceptation du changement » plutôt que « conformité au plan »

Les méthodes Agile

❖ **Les 6 concepts du manifeste Agile, qui doivent être considérés de façon interdépendante:**

- **Valeur**
- **Incertitude**
- **Client**
- **Individus**
- **Equipe**
- **Contexte**

Les méthodes Agile

- ❖ **Vous ne pouvez créer de la valeur si vous ne collaborez pas avec le client, qui précisément définit ce qu'est sa valeur**
- ❖ **Vous ne pouvez attendre d'une équipe qu'elle soit performante si vous ne reconnaissez pas la contribution des individus qui la composent**
- ❖ **Vous ne pouvez maîtriser l'incertitude si vous ne prenez pas en considération la spécificité du contexte**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 1**
 - **Notre priorité est de satisfaire le client en lui livrant très tôt et régulièrement des versions opérationnelles de l'application, source de valeur**
 - **Développement itératif, livraison intermédiaire**
 - **Validation par le client et feedback**
 - **Conformité de la livraison aux attentes ou prise en compte des remarques et re-priorisation**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 2**
 - **Accepter le changement dans les exigences, même tard dans le cycle de vie, pour garantir la compétitivité du client**
 - **Etat d'esprit de l'équipe Agile**
 - **Capacité à comprendre et apprendre comment satisfaire encore mieux la demande**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 3**
 - **Livrer le plus souvent possible des versions opérationnelles de l'application, à une fréquence allant de 2 semaines à 2 mois**
 - **1 version intermédiaire du produit final, visible et testable satisfait davantage le client qu'une documentation à valider**
 - **Il a la preuve tangible que le projet avance**
 - **Il peut exprimer son point de vue sur le résultat présenté**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 4**
 - **Client et développeurs doivent coopérer quotidiennement tout au long du projet**
 - **Les relations conflictuelles ne font pas partie de l'esprit Agile**
 - **Les relations collaboratives et partenariales basées sur la confiance et le consensus sont préférables**
 - **Le client (ou son représentant) est accessible et disponible, totalement impliqué dans toutes les phases du projet**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 5**
 - **Construire des projets autour d'individus motivés. Leur donner l'environnement et le support dont ils ont besoin et leur faire confiance pour remplir leur mission**
 - **Le facteur clé du succès d'un projet est l'équipe**
 - **Tout obstacle à son bon fonctionnement devra être levé**
 - **Un changement, s'il s'avère nécessaire, sera apporté aux processus, aux outils, à l'environnement, à la composition de l'équipe**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 6**
 - **La méthode la plus efficace de communiquer des informations à une équipe et au sein de celle-ci reste la conversation en face à face**
 - **Par défaut, on privilégie l'oral à l'écrit, pour lever toute ambiguïté et favoriser la rapidité de la compréhension**
 - **Tout ne peut être formalisé par écrit, notamment la “connaissance tacite”, c'est à dire, “l'information informelle”, la culture du projet, détenues par chacun au sein d'une équipe**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 7**
 - **Le fonctionnement de l'application est le premier indicateur d'avancement du projet**
 - **Il n'existe pas d'autre indicateur plus pertinent que le pourcentage ou le nombre d'exigences satisfaites**
 - **On ne mesure pas un projet à la quantité de documents produits ou au nombre de lignes de code, non significatifs pour le client**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 8**
 - **Les méthodes Agile recommandent que le projet avance à un rythme soutenable**
 - **La qualité du travail fourni dépend du rythme du travail qui doit être adapté en fonction des spécificités du projet**
 - **Le rythme doit être soutenu et soutenable sur la durée du projet**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 9**
 - **Sponsors, développeurs et utilisateurs devraient pouvoir maintenir un rythme constant indéfiniment**
 - **Ce rythme de travail est à déterminer par l'ensemble des membres de l'équipe et par le client, en fonction de la productivité de l'équipe et des priorités du client**
 - **Travailler en heures supplémentaires pour corriger des bogues ou des régressions n'apporte aucune valeur ajoutée**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 10**
 - **Porter une attention continue à l'excellence technique et à la conception améliore l'agilité**
 - **Maintenir un code propre, évolutif et performant est un objectif permanent de l'équipe**
 - **Il ne s'agit pas de produire du code non exploitable par les autres, ni du jetable**
 - **Cela évite d'enliser les développements ultérieurs, avec des modifications cassant un développement fragile, nécessitant des interventions à des endroits variés du code**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 11**
 - **La simplicité (art de maximiser la quantité de travail non fait) est essentielle**
 - **La simplicité garantit l'évolutivité du système**
 - **La complexité, au contraire coûte davantage et rend plus difficiles les évolutions inhérentes au développement incrémental**
 - **La conception doit ne comporter que des éléments utiles**

Les méthodes Agile

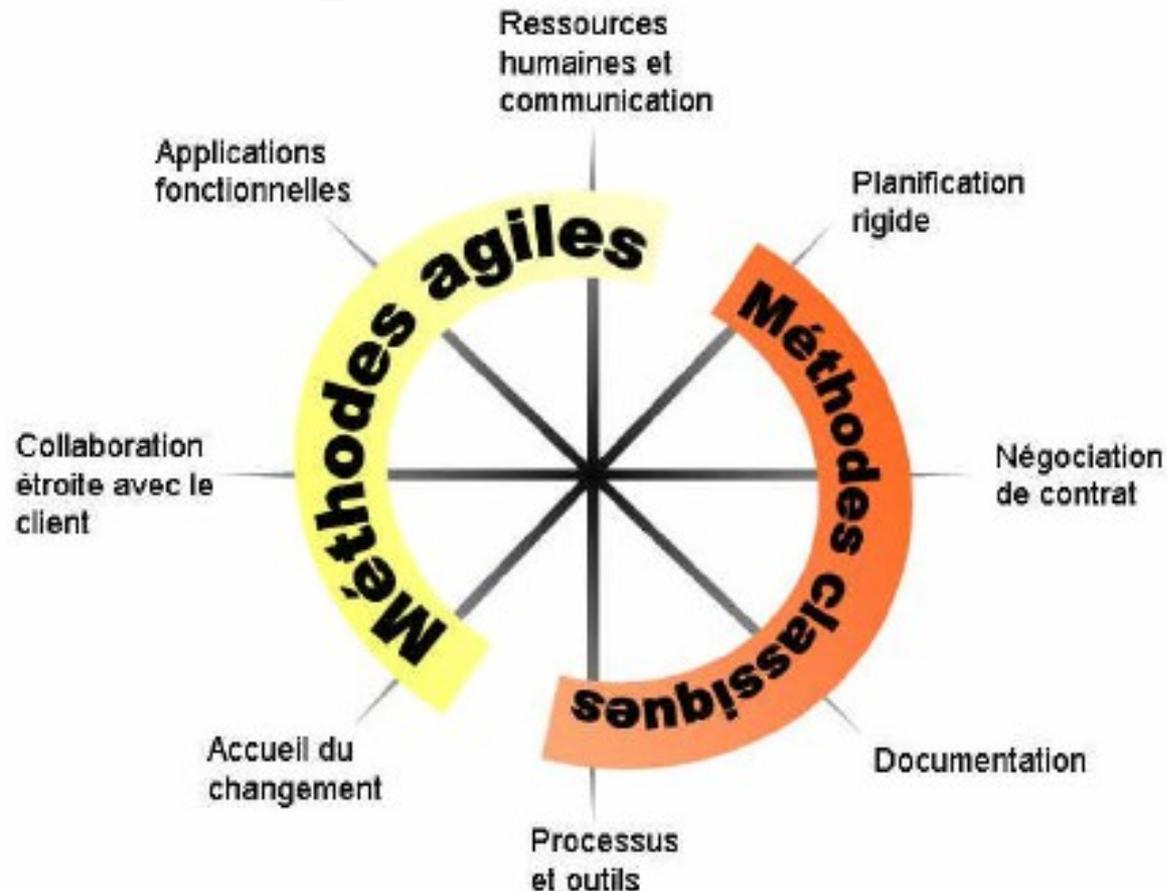
- ❖ **Les 13 principes du manifeste Agile: 12**
 - **Les meilleures architectures, spécifications et conceptions sont le fruit d'équipes qui s'auto-organisent**
 - **Le chef de projet n'est pas celui qui attribue des tâches**
 - **L'équipe, elle-même, se responsabilise et définit ses travaux à réaliser**
 - **Le partage des tâches s'effectue sur la base du volontariat**

Les méthodes Agile

- ❖ **Les 13 principes du manifeste Agile: 13**
 - **A intervalles réguliers, l'équipe s'interroge sur la manière de devenir encore plus efficace, puis ajuste son comportement en conséquence**
 - **L'environnement d'un projet n'est pas constant**
 - **L'équipe s'interroge en permanence sur la façon d'améliorer son fonctionnement afin de s'adapter aux nouvelles conditions**
 - **C'est aussi l'acceptation du changement!**

Les méthodes Agile

❖ Méthodes Agile vs Méthodes classiques



Les méthodes Agile

- ❖ **Approche itérative et incrémentale**
- ❖ **Esprit collaboratif**
- ❖ **Formalisme léger**
- ❖ **Produit de haute qualité**
- ❖ **Acceptation du changement**

Les méthodes Agile

❖ Approche itérative et incrémentale

- Découper le projet en plusieurs étapes d'une durée de quelques semaines: les itérations
- Au cours d'une itération, une version minimale du produit est développée puis soumise au client pour validation
- Les fonctionnalités sont ainsi intégrées au fur et à mesure du cycle de vie sur un mode incrémental

Les méthodes Agile

❖ Approche itérative et incrémentale

- Le système s'enrichissant progressivement pour atteindre les niveaux de satisfaction et de qualité requis
- Chaque itération est un mini-projet en soi
 - Activités de développement menées en parallèle
 - Analyse, conception, codage et test
 - Activités de management de projet

Les méthodes Agile

❖ Approche itérative et incrémentale

- L'objectif est d'atteindre à la fin de chaque itération un sous-ensemble opérationnel du système cible
- Et au terme de la dernière itération, la version finale du produit
- Le résultat d'une itération n'est pas un prototype ou une « proof of concept » mais bien une version intermédiaire du produit final

Les méthodes Agile

❖ Approche itérative et incrémentale

- Les itérations se succèdent et ne peuvent être parallélisées
 - Avancer prudemment et s'adapter au fur et à mesure
 - Pas de plan de management de projet unique mais liste de besoins macroscopiques et macroplanning initial (grandes échéances et jalons principaux)

Les méthodes Agile

❖ Approche itérative et incrémentale

- Les itérations se succèdent et ne peuvent être parallélisées
 - A chaque itération, on sélectionne les fonctionnalités qui seront détaillées puis développées, en fonction de leur priorité
 - Et on établit le microplanning correspondant aux activités nécessaires pour le développement de ces fonctionnalités

Les méthodes Agile

❖ Approche itérative et incrémentale

- Les itérations se succèdent et ne peuvent être parallélisées
- Principe du timebowing (boîte de temps ou tranche de temps)
 - Date d'échéance fixe immuable pour l'itération
 - Mobilisation des efforts sur des objectifs clairs à court terme
 - Si les objectifs ne sont pas atteints, les enseignements seront tirés lors du bilan de l'itération pour corriger les conditions de l'itération suivante

Les méthodes Agile

❖ Approche itérative et incrémentale: avantages

- La communication est de meilleure qualité
- La visibilité est meilleure
- La qualité est évaluée en continu
- Les risques sont détectés très tôt
- L'équipe prend confiance
- Les coûts sont contrôlés

Les méthodes Agile

- ❖ **Approche itérative et incrémentale**
- ❖ **Esprit collaboratif**
- ❖ **Formalisme léger**
- ❖ **Produit de haute qualité**
- ❖ **Acceptation du changement**

Les méthodes Agile

❖ Esprit collaboratif

- Placer les individus et leurs interactions au centre du dispositif
- Communication entre els différents acteurs d'un projet, aus ein de l'équipe, mais aussi entre l'équipe et le client / les utilisateurs

Les méthodes Agile

❖ Esprit collaboratif

- Respect des opinions des autres
- Capacité à exprimer des opinions différentes de façon non agressive
- Aptitude à rechercher et atteindre les consensus sans frustration
- Prédilection à l'autodiscipline, voire à l'autogestion
- Créativité et performance de l'équipe

Les méthodes Agile

- ❖ **Approche itérative et incrémentale**
- ❖ **Esprit collaboratif**
- ❖ **Formalisme léger**
- ❖ **Produit de haute qualité**
- ❖ **Acceptation du changement**

Les méthodes Agile

- ❖ **Approche itérative et incrémentale**
- ❖ **Esprit collaboratif**
- ❖ **Formalisme léger**
- ❖ **Produit de haute qualité**
- ❖ **Acceptation du changement**

Les méthodes Agile

❖ Formalisme léger

- **L'essentiel: les versions intermédiaires du produit**
- **Quelques livrables à produire**
- **Quelques rôles définis**
- **Quelques étapes**
- **Quelques réunions**
- **Quelques outils simples**
- **... et la démarche est formalisée**

Les méthodes Agile

- ❖ **Approche itérative et incrémentale**
- ❖ **Esprit collaboratif**
- ❖ **Formalisme léger**
- ❖ **Produit de haute qualité**
- ❖ **Acceptation du changement**

Les méthodes Agile

❖ **Produit de haute qualité**

- **Sélection des fonctionnalités à implémenter en premier et livraison de valeur en continu**
- **Feedback permanent recueilli auprès du client**
- **Tout défaut peut être détecté et corrigé immédiatement**
- **Refactoring: micro-évolutions ou nettoyage quotidien du code**
- **Approche adaptative**
- **Normes de codage partagées**

Les méthodes Agile

- ❖ **Approche itérative et incrémentale**
- ❖ **Esprit collaboratif**
- ❖ **Formalisme léger**
- ❖ **Produit de haute qualité**
- ❖ **Acceptation du changement**

Les méthodes Agile

❖ Acceptation du changement

- Une équipe Agile se dote de pratiques et d'outils lui facilitant l'accueil du changement

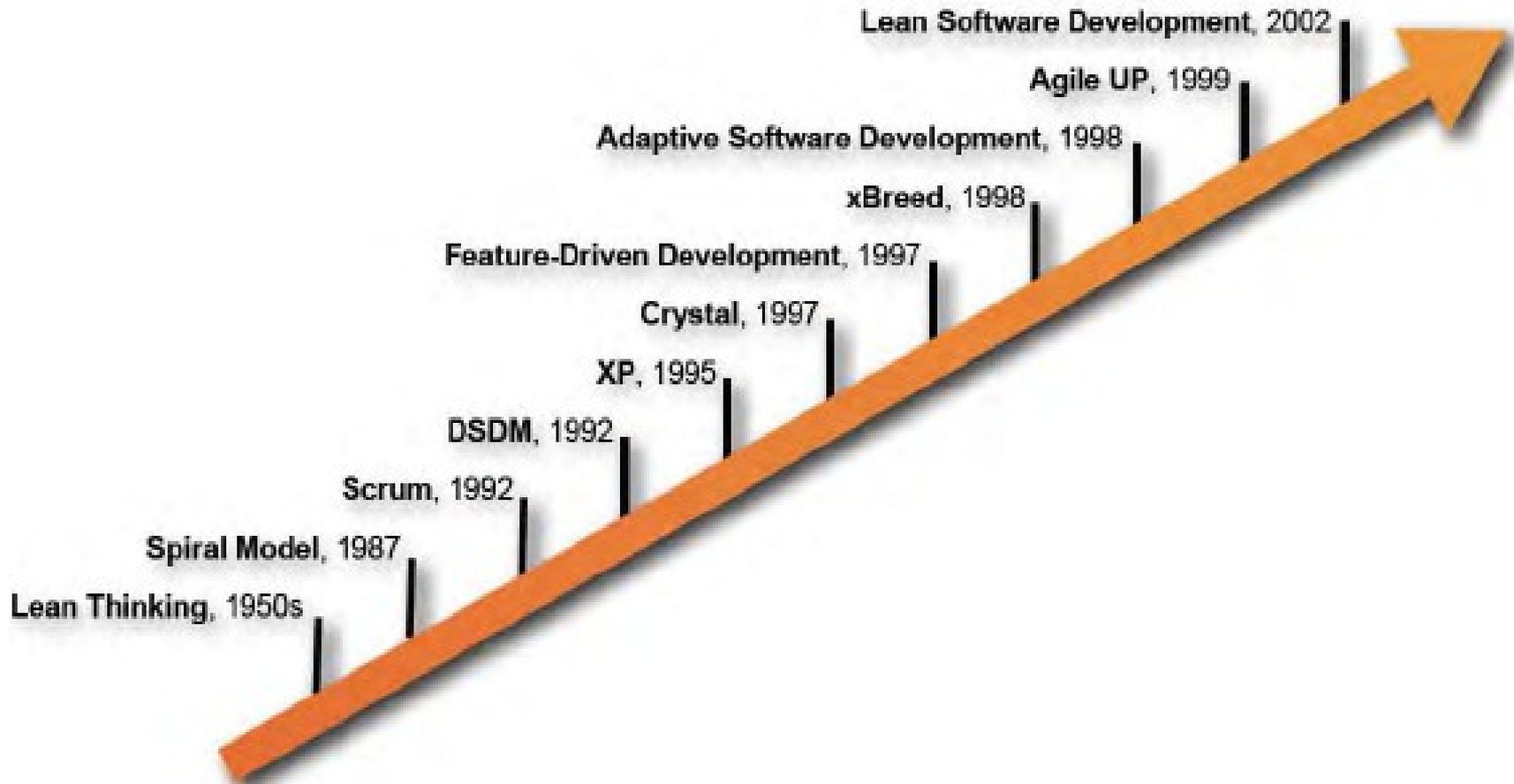
Principales méthodes Agile

- ❖ **ASD (Adaptative Software Development)**
- ❖ **Crystal**
- ❖ **DSDM (Dynamic Software Development Method)**
- ❖ **RAD (Rapid Application Development)**
- ❖ **Lean**
- ❖ **Scrum**

Principales méthodes Agile

- ❖ **UP (Unified Process)**
- ❖ **RUP (Rational Unified Process)**
- ❖ **XP (eXtreme Programming)**

Principales méthodes Agile



Principales méthodes Agile

❖ **ASD (Adaptative Software Development)** **par Jim HIGHSMITH**

- **Le cycle de vie d'un projet ASD se déroule autour d'une série de cycles en 3 volets ou phases clés**
 - **Spéculation**
 - **Collaboration**
 - **Apprentissage**

Principales méthodes Agile

❖ ASD (Adaptative Software Development)

– Spéculation

- Initier le projet (mission, contraintes, collaborateurs, expression des exigences, identification des risques, ...)
- Déterminer la durée du projet, le nombre d'itérations et els dates associées (4 à 8 semaines / itération)
- Affecter un objectif à chaque itération
- Dresser une liste des tâches à réaliser

Principales méthodes Agile

❖ ASD (Adaptative Software Development)

– Collaboration

- Livraison des composants
- Communication forte et assez informelle

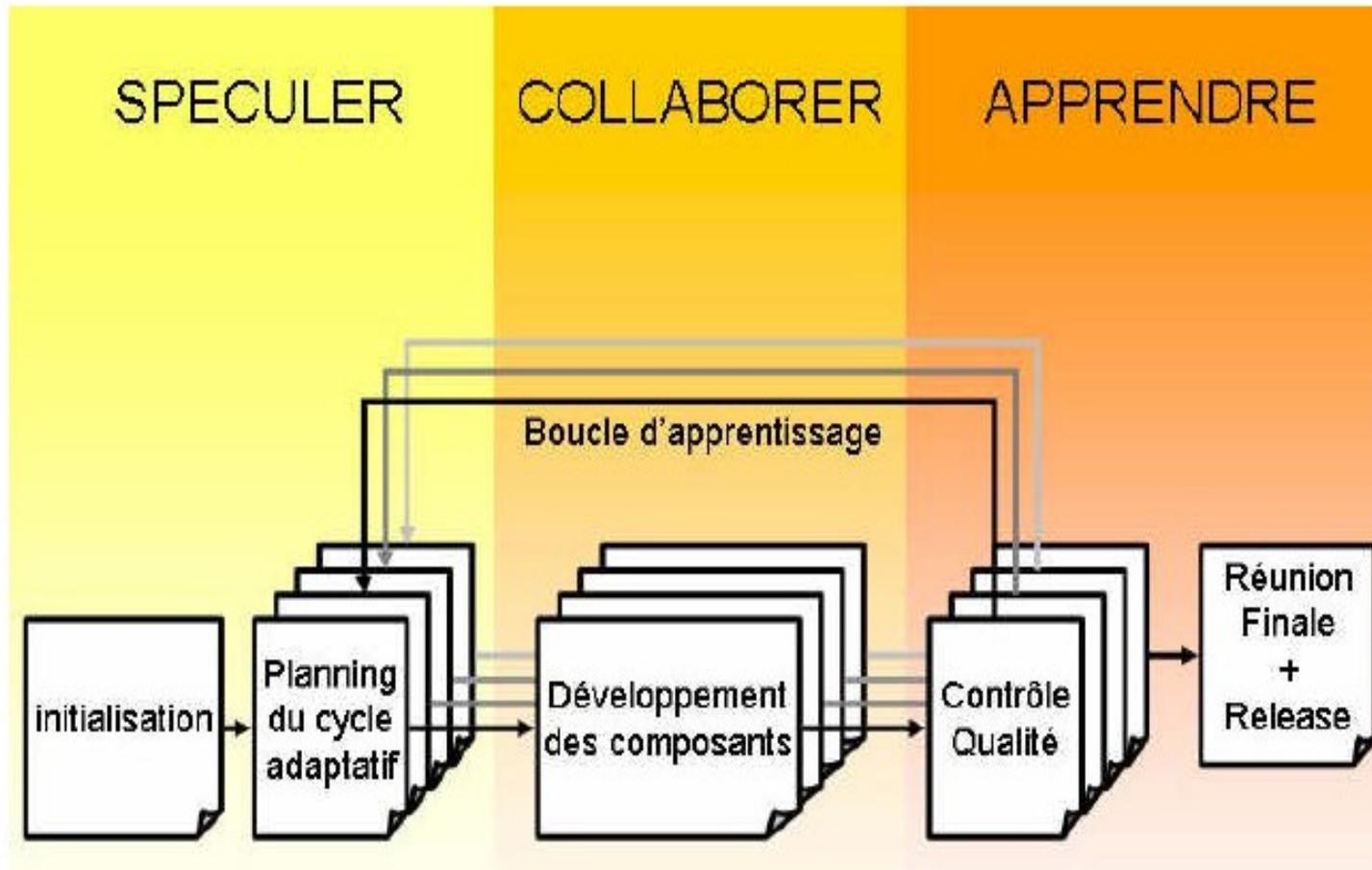
Principales méthodes Agile

❖ ASD (Adaptative Software Development)

- **Apprentissage**
 - **Contrôle qualité**
 - **Suivi et bilan d'avancement**
 - **Communication forte et assez informelle**

Principales méthodes Agile

❖ ASD (Adaptative Software Development)



Principales méthodes Agile

❖ ASD (Adaptative Software Development)

– Caractéristiques principales

- Focaliser sur l'objectif (mission focused)
- Se baser sur des composants (component-based)
- Itérer
- Découper le temps et fixer des deadlines (timeboxing)
- Piloter le projet par les risques (risk-driven development)
- Accepter le changement

Principales méthodes Agile

- ❖ **ASD (Adaptative Software Development)**
 - Ce cycle de vie permet un apprentissage et une adaptation permanente aux évolutions du projet

Principales méthodes Agile

❖ Crystal par Alistair COCKBURN

- Ou plus précisément la famille de méthodologies Crystal
 - Le principe est de sélectionner la méthode applicable en fonction de la criticité du projet et du nombre de personnes à coordonner

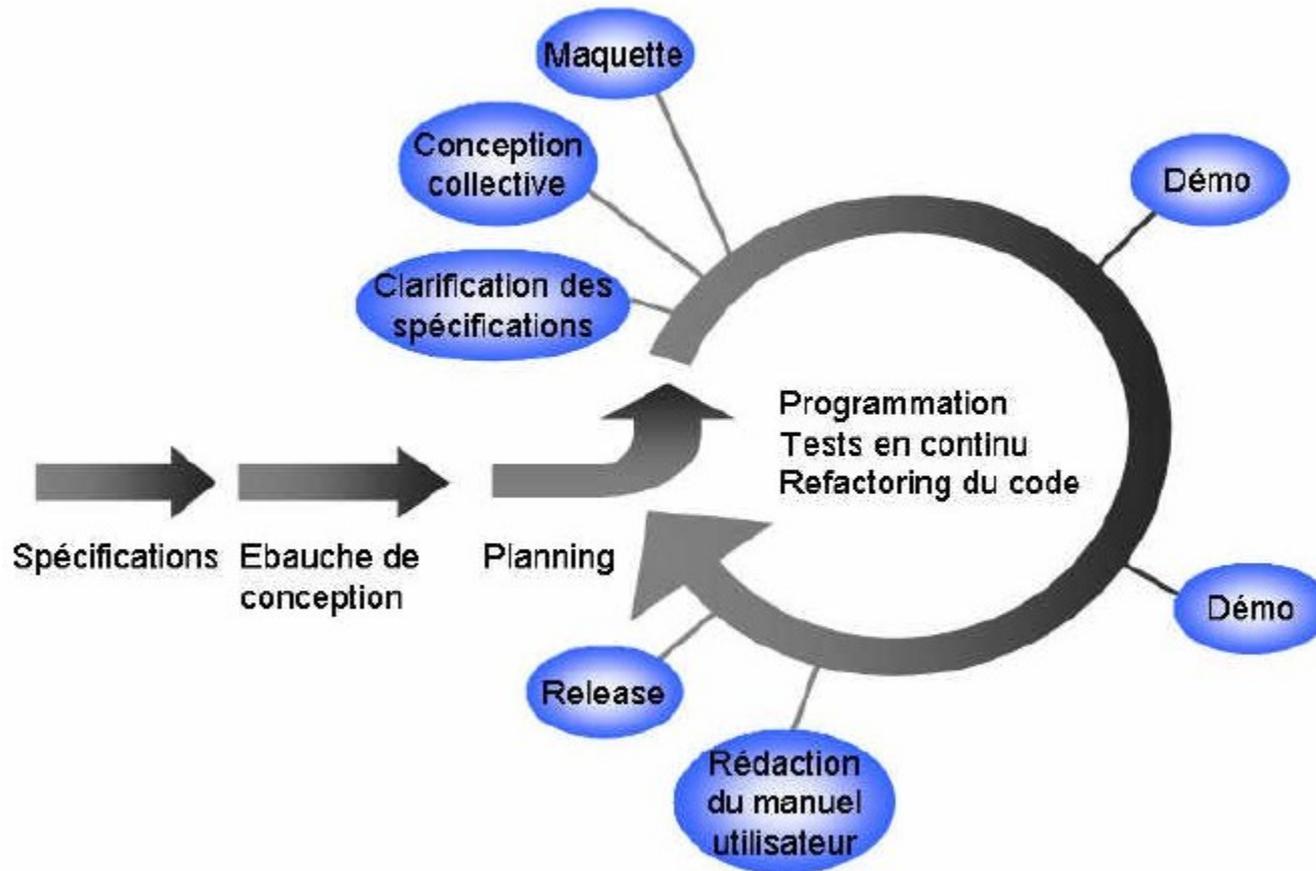
Principales méthodes Agile

❖ Crystal

- Importance de la Communication et du feed-back client
- Releases fréquentes
- Deux grandes phases
 - Conception et planning
 - Itérations
- Adapté à des équipes de 6 personnes maximum (pas de leadership clairement exprimé)

Principales méthodes Agile

❖ Crystal



Principales méthodes Agile

❖ Crystal

- Livraisons fréquentes
- Aménagements permanents
- Communication osmotique
- Confiance, liberté d'expression et sécurité personnelle
- Focus sur l'objectif et disponibilité
- Contact permanent avec les utilisateurs
- Environnement de travail approprié
- Collaboration étroite
- Réflexion constante

Principales méthodes Agile

❖ **DSDM (Dynamic Software Development Method): les principes**

- implication active des utilisateurs
- équipes autorisées à prendre des décisions
- produit rendu tangible aussi souvent que possible
- L'adéquation au besoin métier est le critère essentiel pour l'acceptation des fournitures
- Un développement itératif et incrémental permet de converger vers une solution appropriée

Principales méthodes Agile

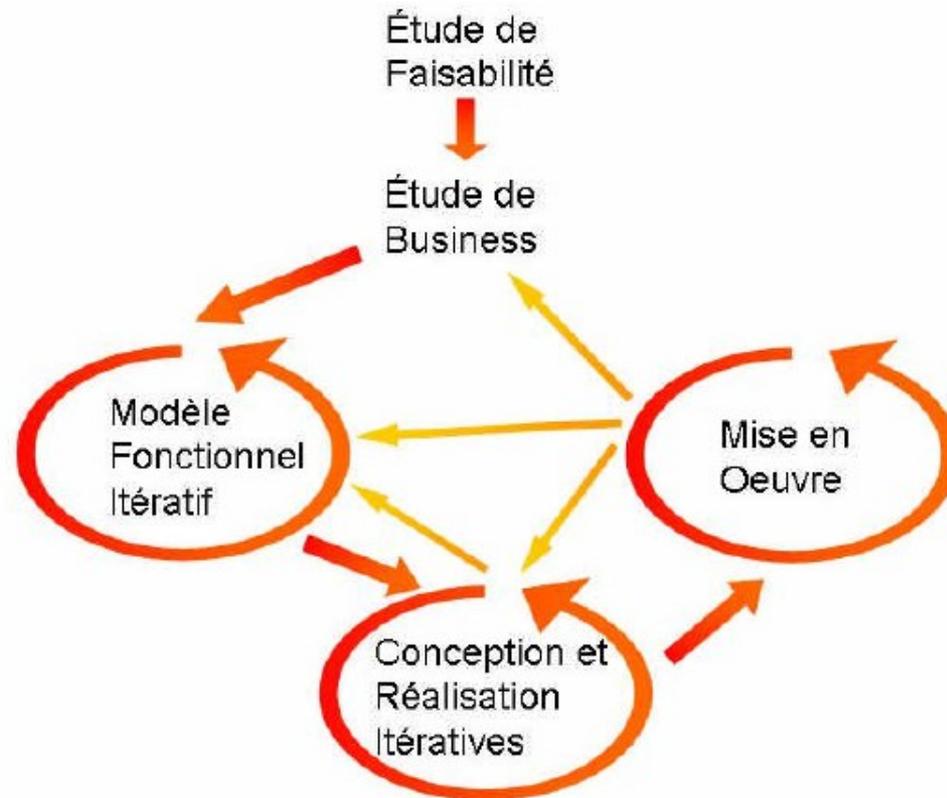
❖ DSDM (Dynamic Software Development Method): les principes

- Toute modification pendant la réalisation est réversible
- besoins définis à un niveau de synthèse
- tests intégrés pendant tout le cycle de vie
- esprit de coopération entre tous



Principales méthodes Agile

❖ DSDM (Dynamic Software Development Method): le cycle de vie



Principales méthodes Agile

❖ DSDM (Dynamic Software Development Method)

– L'étude de faisabilité

- Bien définir le problème à résoudre
- En étudier la faisabilité (technique, méthodologique, budgétaire, ...)

Principales méthodes Agile

❖ DSDM (Dynamic Software Development Method)

– L'étude du business

- Déterminer et analyser les processus métier qui doivent être automatisés et les besoins en informations
- Grâce à des ateliers facilités, les utilisateurs définissent leurs besoins
- Définition de l'architecture globale
- Ecriture du plan global de prototypage

Principales méthodes Agile

❖ **DSDM (Dynamic Software Development Method)**

– **Le modèle fonctionnel itératif**

- **Ce modèle produit décrit les besoins en détail et permet de définir quand et comment ils seront satisfaits**
- **Le résultat est une série de modules logiciels constituant un prototype fonctionnel**

Principales méthodes Agile

❖ **DSDM (Dynamic Software Development Method)**

- **La conception et les développement itératifs**
 - **Fournir un système intégrant toutes les fonctionnalités, conforme aux besoins définis**

Principales méthodes Agile

❖ DSDM (Dynamic Software Development Method)

– La mise en oeuvre

- Phase de livraison
- Prise en main de l'application par les utilisateurs
- Formation des utilisateurs
- Test de l'application par les utilisateurs
- Contrôle de la qualité de la documentation avant mise en production
- Bilan et capitalisation des bonnes pratiques mises en oeuvre

Principales méthodes Agile

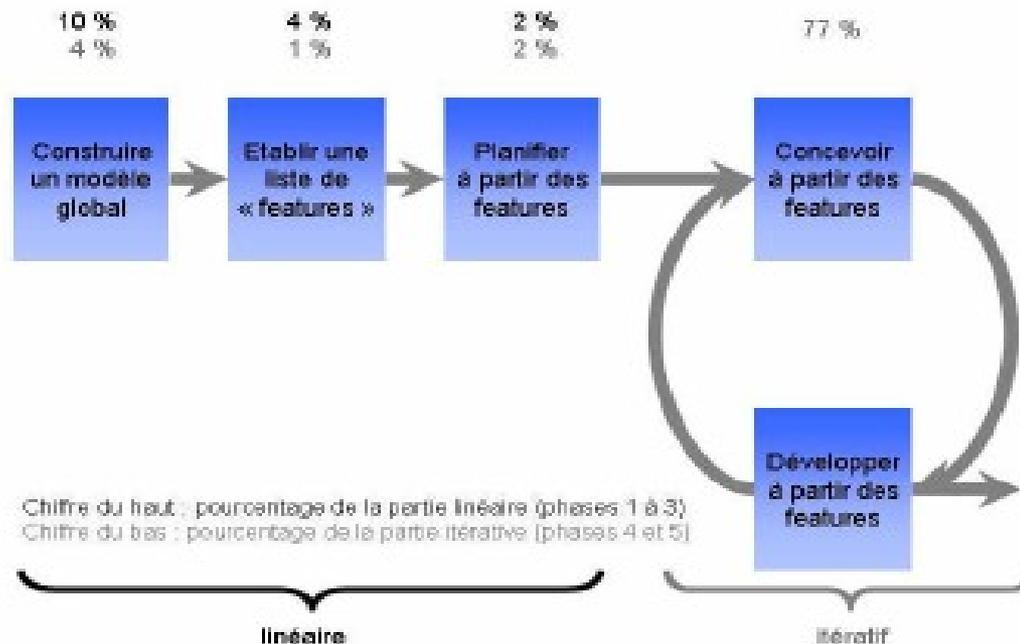
❖ DSDM (Dynamic Software Development Method): les rôles

- Sponsor exécutif
- Visionnaire : **Expert métier**
- Utilisateur ambassadeur
- Utilisateur conseiller
- Chef de projet
- Coordinateur technique
- Chef d'équipe
- Développeur
- Facilitateur
- Rapporteur

Principales méthodes Agile

❖ FDD (Feature Driven Development)

- Méthode créée par Jeff de Luca et Peter Coad
 - Itérations très courtes
 - Spécifications découpables en features, regroupables en features set
 - Le feature est une fonctionnalité porteuse de valeur pour le client



Principales méthodes Agile

❖ FDD (Feature Driven Development)

- Phase initiale : développer un modèle global
 - Constitution l'équipe de modélisation
 - Etude du modèle (formation des développeurs)
 - Etude documentaire
 - Elaboration d'une liste informelle de features
 - Modélisation de classes en petits groupes puis consolidation

- Etablissement de la liste détaillée de features
 - A partir de la base informelle, listing détaillé des features
 - Les features trop complexes sont éclatés
 - Les features sont triés par ordre de priorité
 - L'ensemble est validé

- Planification à partir des features
 - Constitution de l'équipe planning
 - Séquencement des features
 - Affectation des classes
 - Affectation des features aux développeurs seniors

Principales méthodes Agile

❖ FDD (Feature Driven Development)

- Conception à partir des features
 - Formation de groupes pour la réalisation de chaque feature
 - Etude des documents de référence associés au feature
 - Construction des diagrammes de séquence
 - Conception détaillée des classes
 - Inspection de l'ensemble

- Construction à partir des features
 - Implémentation des classes / méthodes
 - Inspection du code + tests unitaires
 - Préparation pour l'intégration

Principales méthodes Agile

❖ FDD (Feature Driven Development)

Principales méthodes Agile

- ❖ **RAD (Rapid Application Development)**
 - N'est pas réellement une méthode Agile mais en est à l'origine, car première approche semi-itérative incrémentale

Principales méthodes Agile

❖ RAD (Rapid Application Development)

– 5 phases:

- Initialisation
- Cadrage
- Design
- Construction
- Finalisation

Principales méthodes Agile

❖ RAD (Rapid Application Development)

– 5 phases:

- Initialisation
- Cadrage
- Design
- Construction
- Finalisation

Principales méthodes Agile

❖ RAD (Rapid Application Development)

– Initialisation

- Définition du périmètre
- Organisation du travail par thème
- Définition des ressources nécessaires

– Cadrage

- Phase d'expression des besoins

Principales méthodes Agile

❖ RAD (Rapid Application Development)

– Design

- Phase de conception avec modélisation du système

– Construction

- L'équipe ou « SWAT » (unité d'élite) construit l'application de façon itérative, par module ou par thème, que valident les utilisateurs

– Finalisation

- Livraison globale de l'application

Principales méthodes Agile

❖ Lean (Lean software management)

- Amélioration continue, dans le respect des collaborateurs et partenaires, pour la recherche de l'excellence
 - Challenger sa performance tous les jours
 - Voir les problèmes
 - Résoudre les problèmes
 - En tirer les bons enseignements

Principales méthodes Agile

❖ Lean: les outils

- Flux tendu
- Elimination de la variation, des gaspillages et de la surcharge
- Construire de la qualité
 - Intégration de la qualité dans les processus
- Management visuel
- Standardisation créative
- Formation et mentoring

Principales méthodes Agile

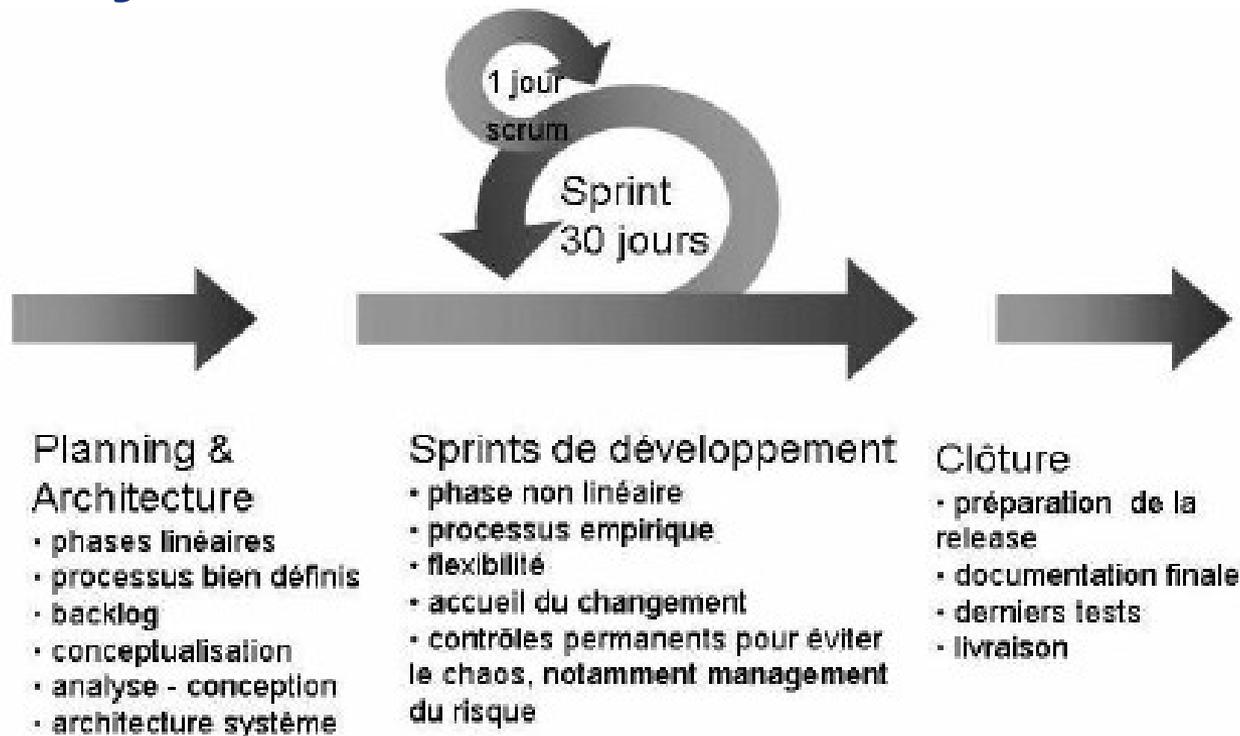
❖ Scrum

- **Développé en 1993 par Ken SCHWABER et par Jeff SUTHERLAND**
- **Cycle de vie organisé en sprints (itérations)**
- **Product backlog: liste des exigences initiales, dressée et hiérarchisée avec le client**

Principales méthodes Agile

❖ Scrum

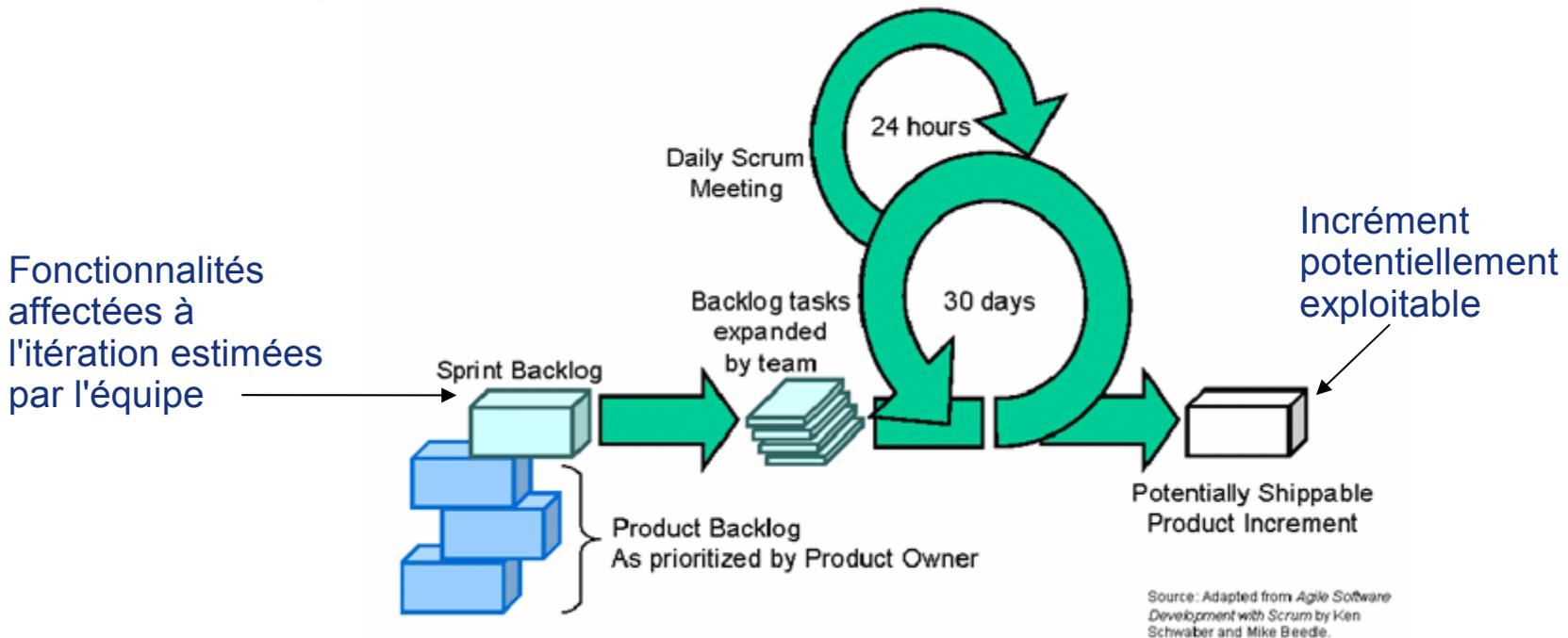
– Le cycle de vie



Principales méthodes Agile

❖ Scrum

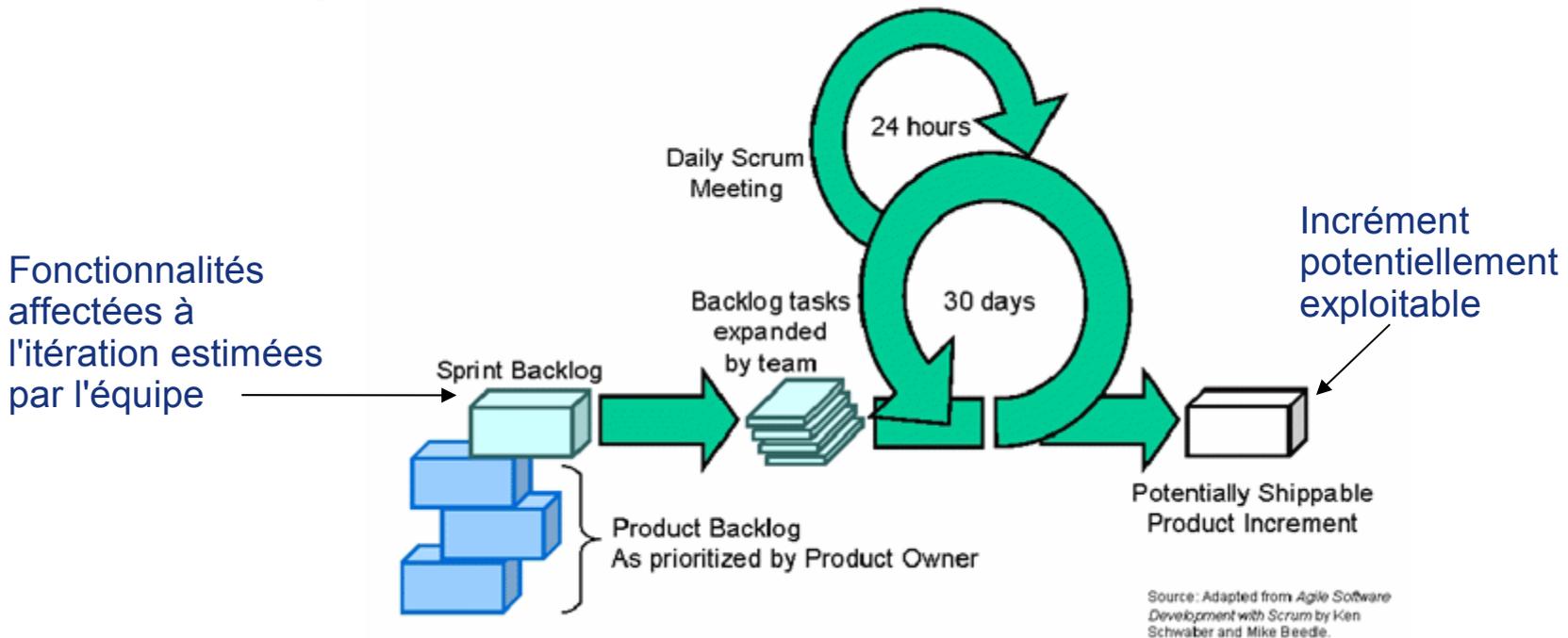
– Le cycle de vie



Principales méthodes Agile

❖ Scrum

– Le cycle de vie



Principales méthodes Agile

❖ Scrum

- Phase initiale
 - Planning
 - Mise en place d'un backlog (liste de tâches à effectuer)
 - Définition de l'équipe
 - Analyse des risques - Budget

- Sprints
 - 30 jours isolés de toute influence extérieure
 - Un sprint backlog est suivi et réalisé
 - Réunion quotidienne Scrum
 - Réunion post-sprint de présentation des résultats

- Clôture
 - documentation finale
 - Livraison...

Principales méthodes Agile

❖ Scrum

– Les rôles

- **Product owner**
- **Scrum master**
- **L'équipe**

Principales méthodes Agile

❖ Scrum

– Les valeurs

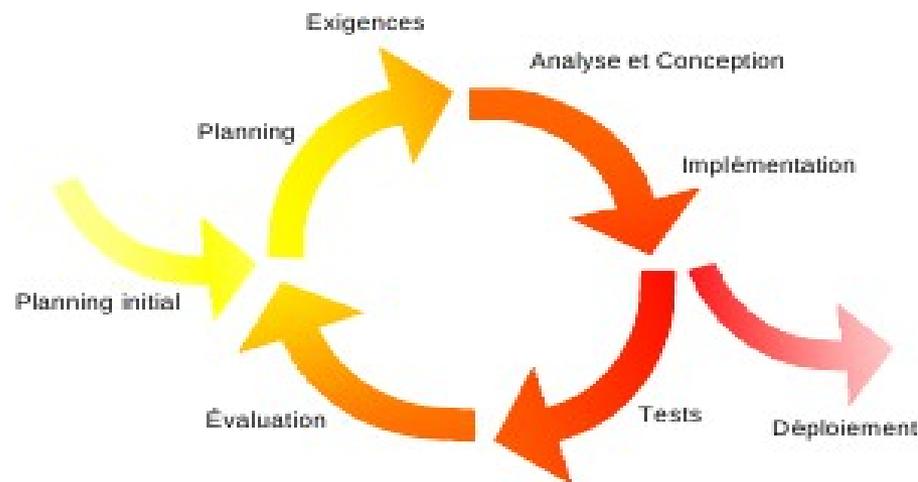
- **Visibilité**
- **Inspection**
- **Adaptation**

Principales méthodes Agile

❖ UP (Unified Process)

Les fondamentaux d'UP

- Pilotage par les cas d'utilisation
- UP est centré sur l'architecture
- UP est itérative et incrémentale
- UP gère les besoins et les exigences
- UP est fondée sur la production de composants
- UP pratique la modélisation visuelle
- UP surveille la qualité et les risques



Principales méthodes Agile

❖ UP (Unified Process)

■ Les activités dans UP

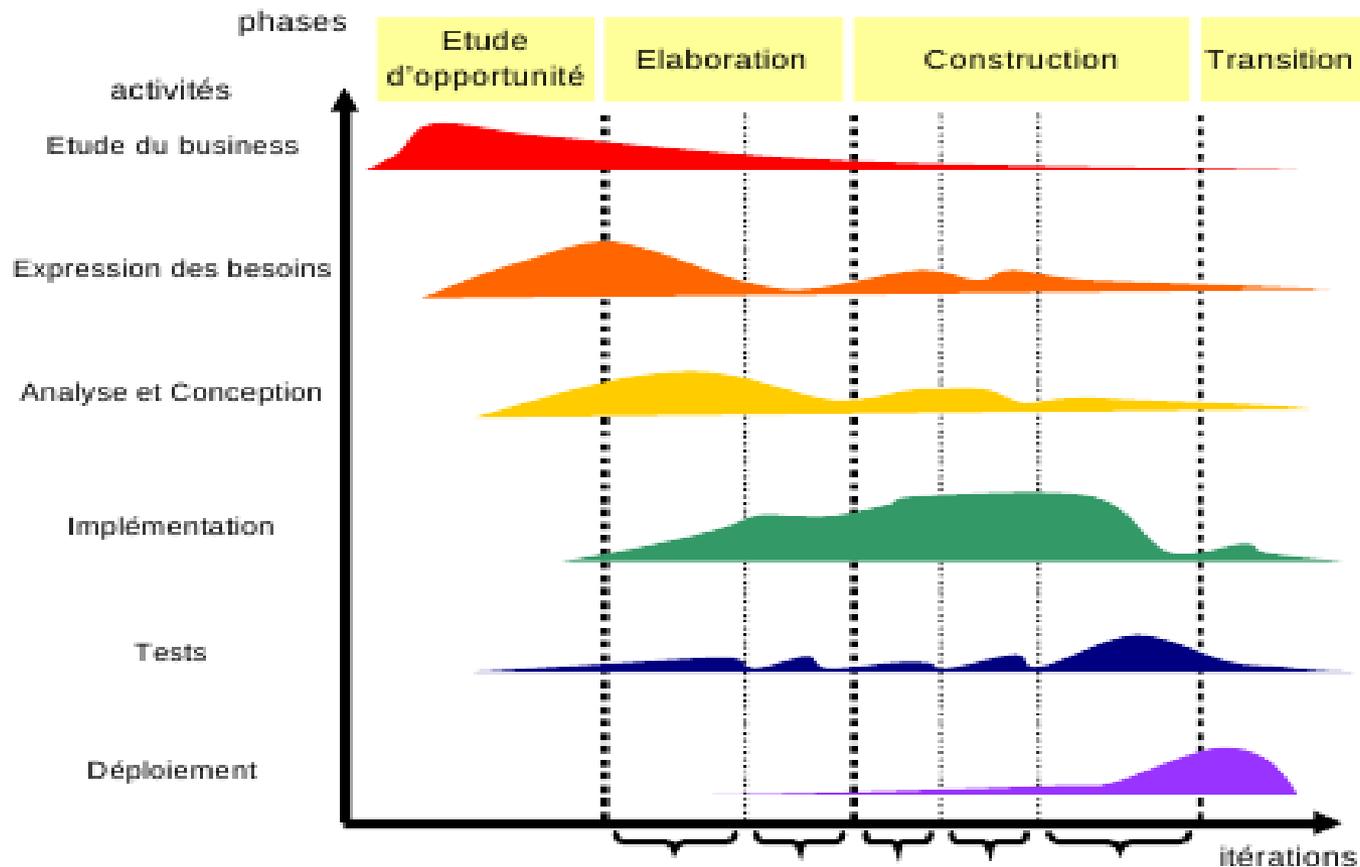
- Expression des besoins
- Analyse (expression des besoins du pt de vue développeur)
- Conception (définition de l'architecture, incrémentale)
- Implémentation
- Tests

■ Les phases du cycle de vie

- Etude d'opportunité (faisabilité, analyse des risques, vision globale des exigences...10% des uses cases cartographiés)
- Elaboration (précision de l'architecture, 80% des use cases cartographiés)
- Construction (fourniture d'une version beta)
- Transition (correction de bugs, préparation de l'itération suivante)

Principales méthodes Agile

❖ UP (Unified Process)



Principales méthodes Agile

❖ UP (Unified Process)

– Les 6 bonnes pratiques

- **Pilotage itératif et incrémental piloté par les risques et les cas d'utilisation**
- **Gestion rigoureuse des exigences**
- **Développement centré sur l'architecture**
- **Modélisation graphique des exigences**
- **Contrôle permanent de la qualité**
- **Contrôle des changements**

Principales méthodes Agile

❖ RUP (Rational Unified Process)

- Adaptation de UP par Rational en 1998, rachetée ensuite par IBM

Principales méthodes Agile

❖ XP (eXtreme Programming)

- Créé en 1996 par Kent BECK et Ron JEFFRIES

Principales méthodes Agile

❖ XP (eXtreme Programming)

– 4 valeurs clés

- Communication
 - rendre la communication omniprésente entre tous les intervenants
- Simplicité
 - il coûte moins cher d'aller au plus simple et de rajouter des fonctionnalités par la suite plutôt que de concevoir dès le départ un système très compliqué dont on risque de n'avoir jamais l'utilité
- Feedback
 - indispensable pour que le projet puisse accueillir le changement
- Courage
 - concerne aussi bien les développeurs que le client

... et respect

Principales méthodes Agile

❖ XP (eXtreme Programming)

– 12 principes

- Feedback rapide
- Assumer la simplicité
- Changements incrémentaux
- Accueillir le changement à bras ouverts
- Un travail de qualité
- Apprendre à apprendre
- Faible investissement au départ
- Jouer pour gagner
- Des expériences concrètes
- Communication ouverte et honnête
- Responsabilités acceptées
- Adaptation aux conditions locales
- Voyager léger
- Mesures honnêtes

Principales méthodes Agile

❖ XP (eXtreme Programming)

– 12 principes déclinés en pratiques

■ Planning game

- phase d'exploration : écriture des besoins sous forme de "user stories" et estimation de leur durée
- phase d'engagement : classement des user stories par ordre de priorité. phase de direction : mise à jour du planning

■ Petites releases

■ Utilisation de métaphores pour décrire l'architecture du système

■ Conception simple : toujours développer la solution la plus simple possible

■ Tests (unitaires et fonctionnels)

Principales méthodes Agile

❖ XP (eXtreme Programming)

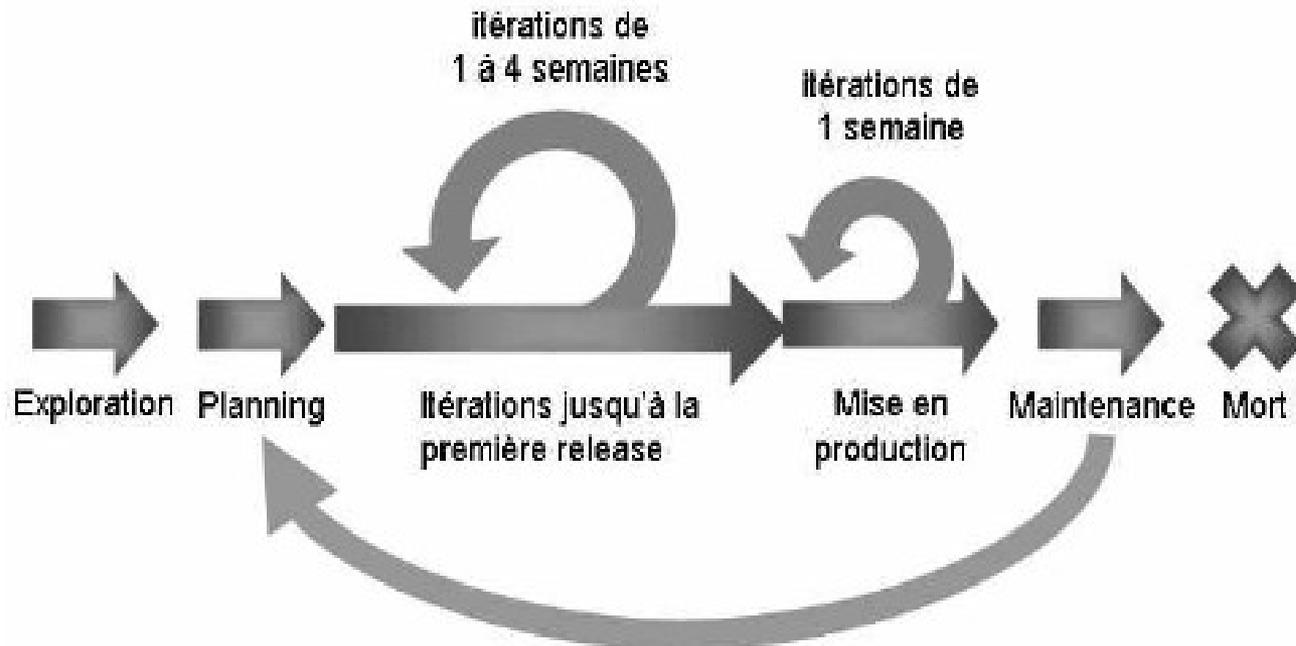
– 12 principes déclinés en pratiques

- Refactoring du code : retravailler le code pour le rendre plus lisible et plus robuste
- Programmation en binôme
- Propriété collective du code
- Intégration continue (plusieurs fois par jour)
- Pas de surcharge de travail : ne pas dépasser 40 heures de travail par semaine
- Client sur site : présence sur site d'une personne minimum à temps plein pendant toute la durée du projet
- Standards de code (normes de nommage et de programmation)

Principales méthodes Agile

❖ XP (eXtreme Programming)

– Le cycle de vie



maintenance = ajouter des fonctionnalités → nouvelles releases
utilise le même processus que pour la release 1

Principales méthodes Agile

❖ XP (eXtreme Programming)

– Les rôles

- **Développeur**
 - **travaille en binôme, communique**
 - **doit être autonome**
 - **a une double compétence : développeur – concepteur**

- **Client**
 - **doit apprendre à exprimer ses besoins sous forme de user stories**
 - **a à la fois le profil de l'utilisateur et une vision plus élevée sur le problème et l'environnement du business**
 - **doit apprendre à écrire les cas de tests fonctionnels**

- **Testeur**
 - **a pour rôle d'aider le client à choisir et à écrire ses tests fonctionnels**

Principales méthodes Agile

❖ XP (eXtreme Programming)

– Les rôles

- **Tracker**
 - aide l'équipe à mieux estimer le temps nécessaire à l'implémentation de chaque user story
 - contrôle la conformité de l'avancement au planning

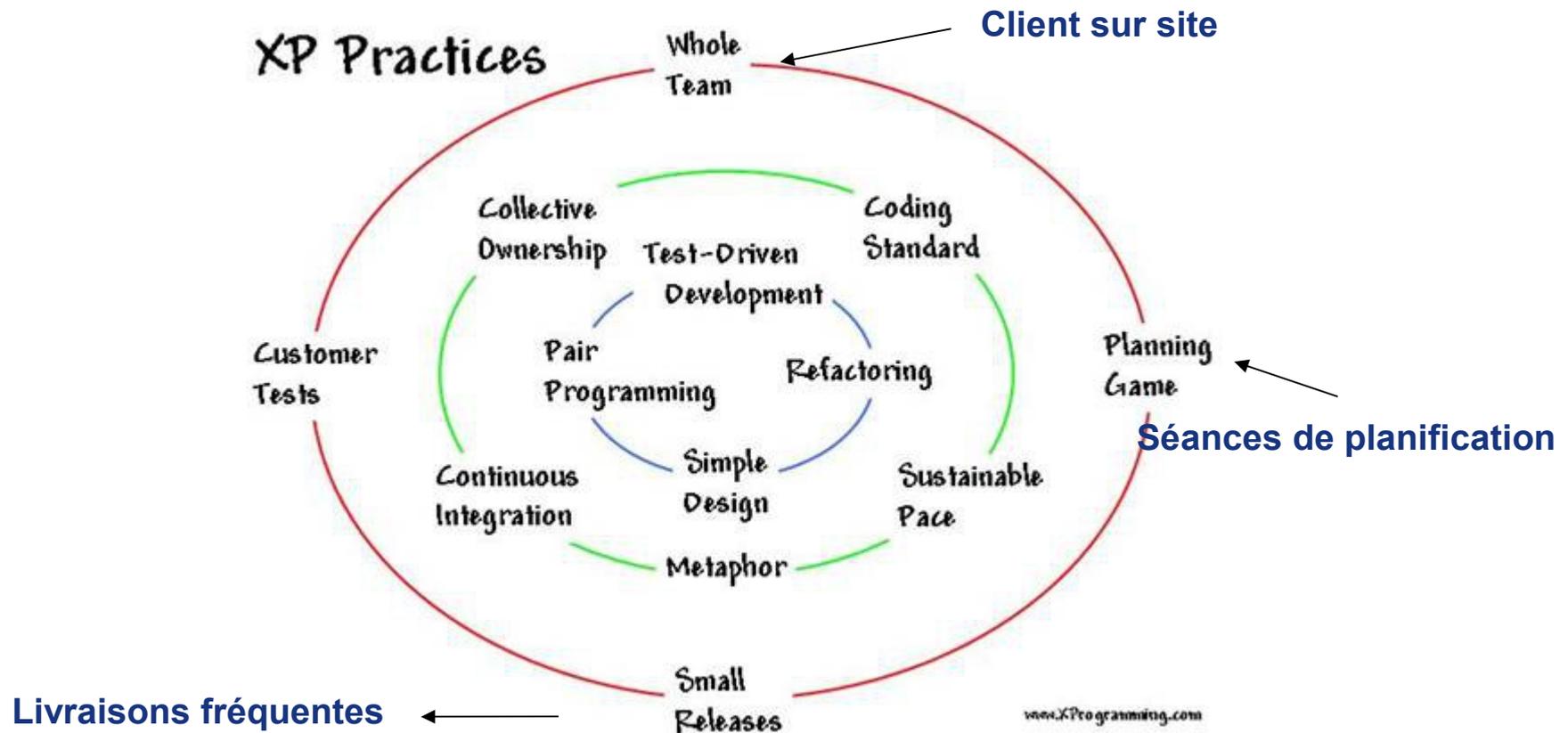
- **Coach**
 - recadre le projet
 - ajuster les procédures
 - doit intervenir de la manière la moins intrusive possible

- **Consultant**
 - n'apporte pas de solution toute faite
 - apporte à l'équipe les connaissances nécessaires pour qu'elle résolve elle-même les problèmes

- **Big Boss**
 - apporte à l'équipe courage et confiance

Principales méthodes Agile

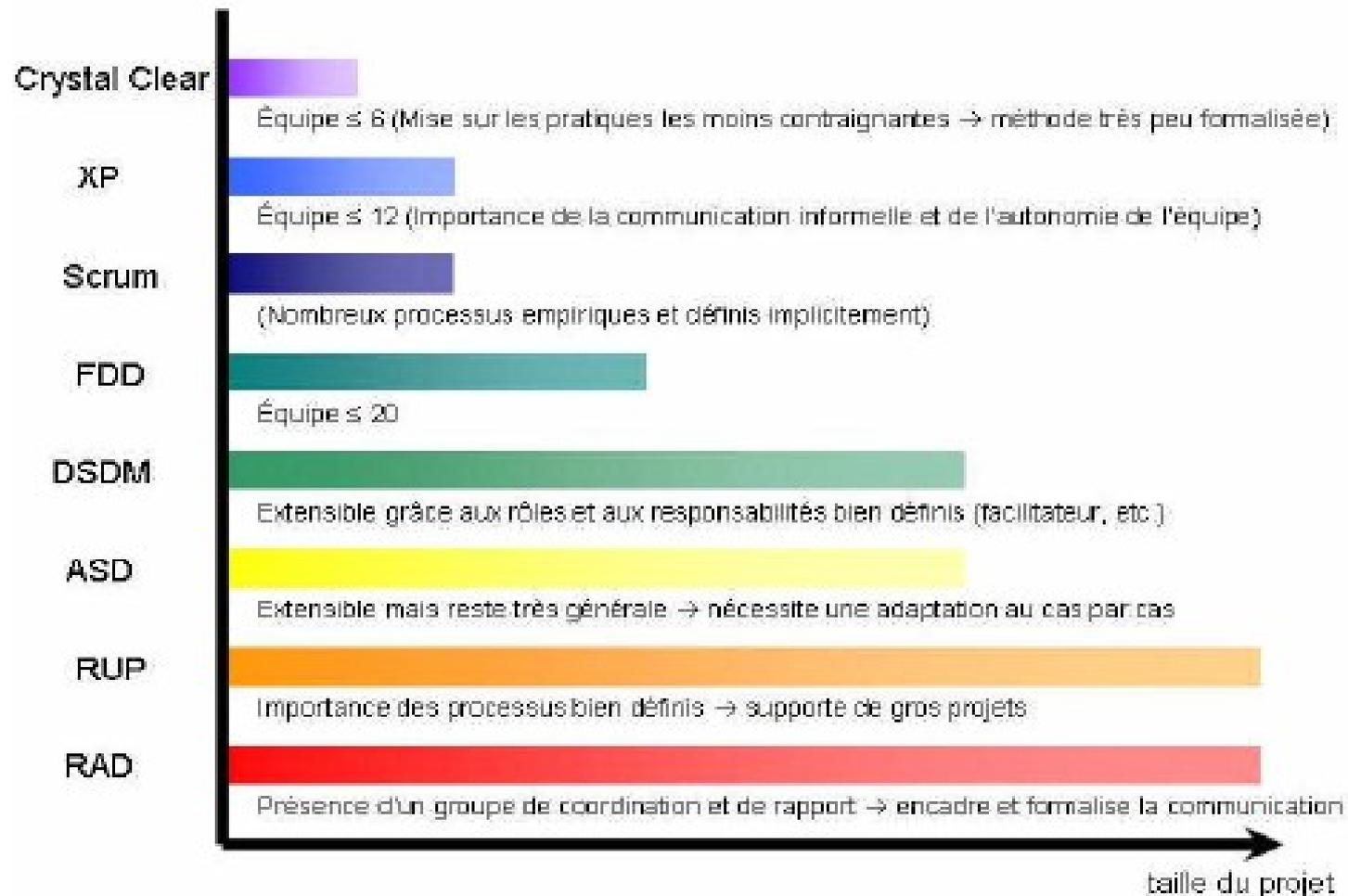
❖ XP (eXtreme Programming)



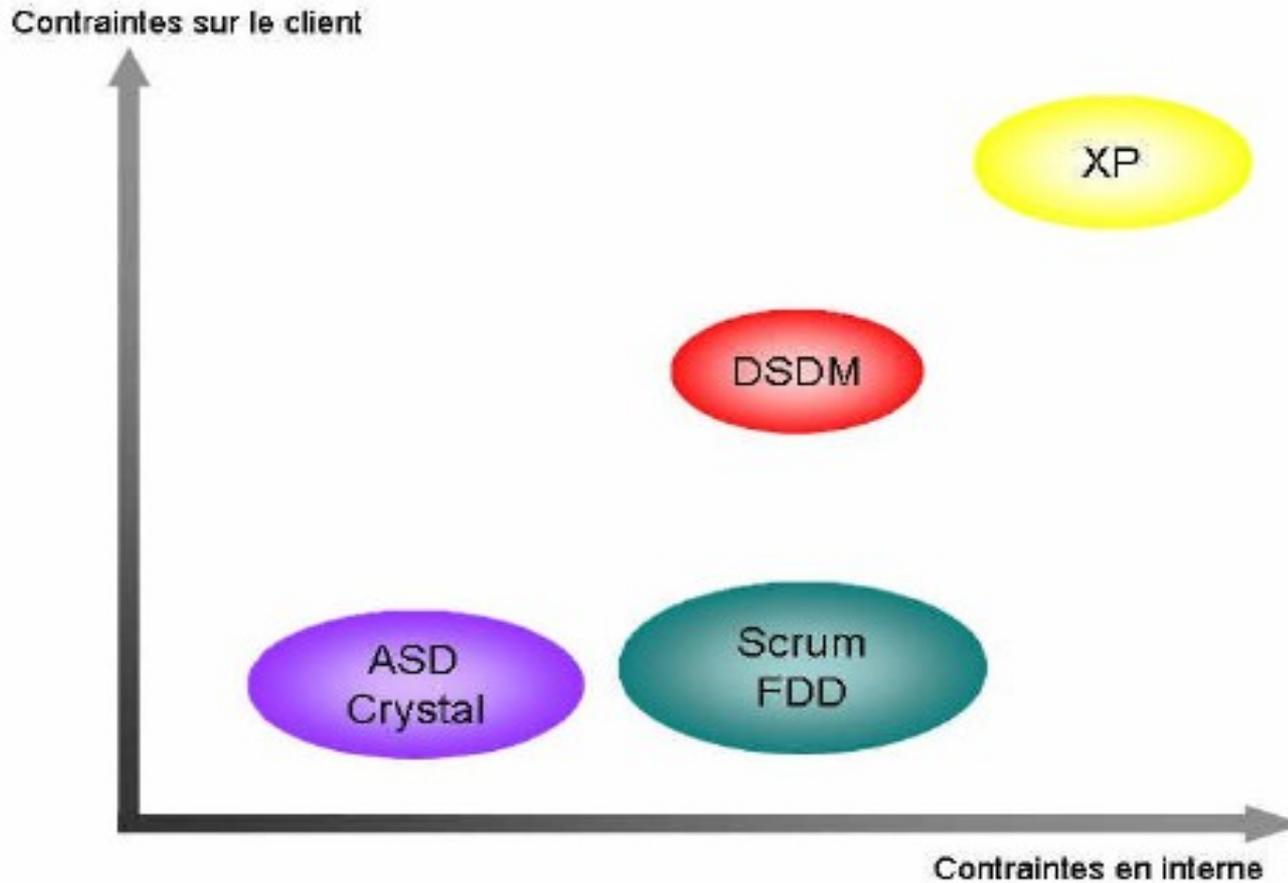
Avantages des méthodes Agile

- ❖ **Apport de valeur ajoutée**
- ❖ **Adaptabilité**
- ❖ **Visibilité**
- ❖ **Réduction des risques**
- ❖ **Qualité logicielle**

Synthèse des méthodes Agile



Synthèse des méthodes Agile



Synthèse des méthodes Agile

- Petites équipes et outils véloces
- Polyvalence des équipes
- Feedback client et itérations
- Gestion des risques
- Collaboration MOA et MOE
- Tester toujours et encore
- Les valeurs clés
 - Humilité
 - Pragmatisme
 - Soucis de la qualité
 - Partage
 - Courage

Synthèse des méthodes Agile

- ❖ **Apport de valeur ajoutée**
- ❖ **Adaptabilité**
- ❖ **Visibilité**
- ❖ **Réduction des risques**
- ❖ **Qualité logicielle**

Différences entre méthodes traditionnelles et méthodes Agile

❖ Cycle de vie

- approche traditionnelle

En cascade ou en V, sans
rétroaction possible

Phases séquentielles

- approche Agile

Itératif et incrémental

Différences entre méthodes traditionnelles et méthodes Agile

❖ Planification

- approche traditionnelle

Prédictive

Plans plus ou moins détaillés

Périmètre et exigences définis et stables

- approche Agile

Adaptative

Plusieurs niveaux de planification (macro et micro)

Ajustements au fil de l'eau en fonction des changements survenus

Différences entre méthodes traditionnelles et méthodes Agile

❖ Documentation

- approche traditionnelle

Produite en quantité importante comme support de communication, validation et contractualisation

- approche Agile

Réduite au strict nécessaire au profit d'incrémentes fonctionnels opérationnels pour obtenir le feedback du client

Différences entre méthodes traditionnelles et méthodes Agile

❖ Equipe

- approche traditionnelle

Une équipe avec des ressources spécialisées dirigée par un chef de projet

- approche Agile

Une équipe responsabilisée où l'initiative et la communication sont privilégiées, soutenue par le chef de projet

Différences entre méthodes traditionnelles et méthodes Agile

❖ Qualité

- approche traditionnelle

Contrôle qualité à la fin du cycle de développement

Le client découvre le produit fini

- approche Agile

Contrôle qualité précoce et permanent, au niveau du produit et du processus

Le client visualise les résultats tôt et souvent

Différences entre méthodes traditionnelles et méthodes Agile

❖ Changement

- approche traditionnelle

Résistance
opposition
voire
Processus lourd de gestion
des changements
acceptés

- approche Agile

Accueil favorable
Intégré dans le processus

Différences entre méthodes traditionnelles et méthodes Agile

❖ Suivi de l'avancement

- approche traditionnelle

Mesure de la conformité
aux plans initiaux
Analyse des écarts

- approche Agile

Un seul indicateur
d'avancement : le
nombre de
fonctionnalités
implémentées et le
travail restant à faire

Différences entre méthodes traditionnelles et méthodes Agile

❖ Gestion des risques

- approche traditionnelle

Processus rigoureux distinct et

- approche Agile

Intégrée dans le dispositif global

Responsabilisation de chacun dans l'identification et la résolution des risques

Pilotage par les risques

Différences entre méthodes traditionnelles et méthodes Agile

❖ **Mesure du succès**

- approche traditionnelle

Respect des engagements initiaux (contenu, coût, délai, qualité)

- approche Agile

Satisfaction client par la livraison de valeur ajoutée

Différences entre méthodes traditionnelles et méthodes Agile

❖ Planification

- approche traditionnelle

En cascade ou en V, sans
rétroaction possible

Phases séquentielles

- approche Agile

Itératif et incrémental

Introduction d'Agile en entreprise

