

Scrum

-

Tour d'horizon de la méthode

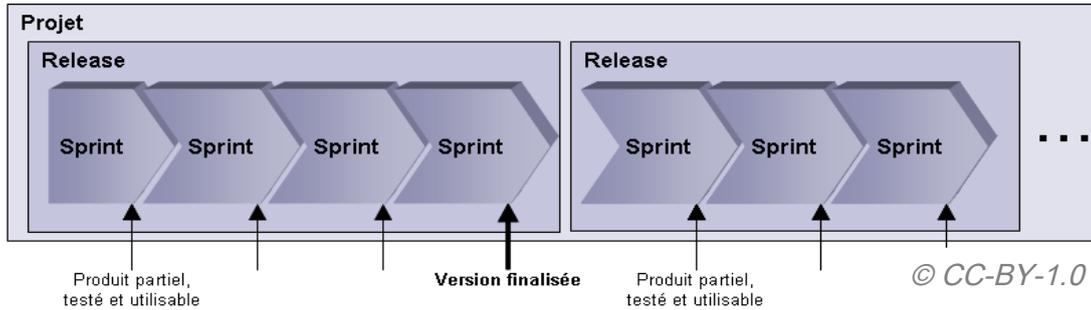
- Agilité
- Scrum
- Pilotage d'un projet agile selon Scrum
- Contractualisation
- Forces & questions ouvertes

- **Méthodes de développement d'applications logicielles caractérisées par**
 - Leur approche adaptative plutôt que prédictive
 - Leur orientation vers les personnes plutôt que vers les processus
 - Méthodes les plus connues : XP, Scrum
- **Concepts formalisés en 2001 par le Manifeste Agile**
- **Les 4 principes essentiels du Manifeste Agile**
 - L'équipe >> Personnes et interactions plutôt que processus et outils
 - L'application >> Logiciel fonctionnel plutôt que documentation complète
 - La collaboration >> Collaboration avec le client plutôt que négociation de contrat
 - L'acceptation du changement >> Réagir au changement plutôt que suivre un plan

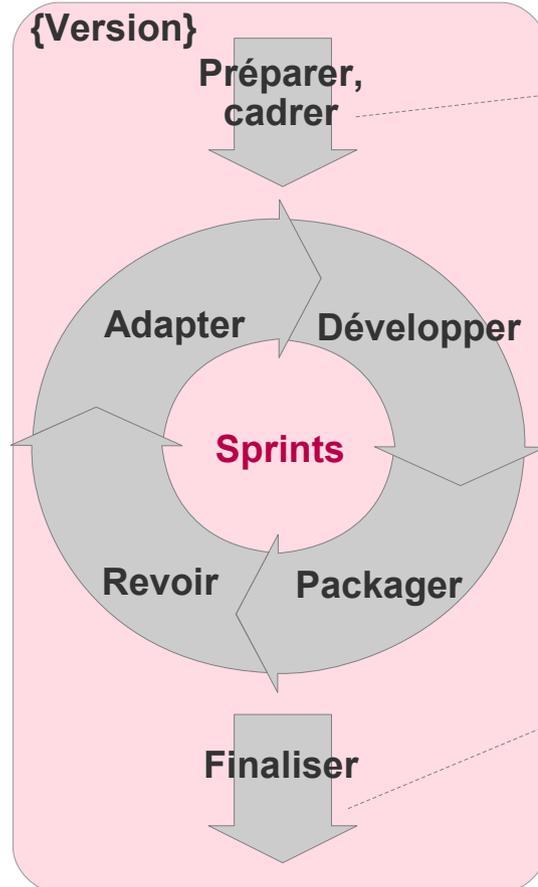
Un objectif : la réussite des projets informatiques

- Initialisée par Ken Schwaber et Jeff Sutherland en 1996 (évolutive)
- Utilisée pour organiser les équipes et améliorer la productivité avec une meilleure qualité
 - L'équipe est impliquée : elle peut choisir la quantité de travail à réaliser et comment le réaliser de façon optimale
 - > climat serein propice à une meilleure productivité
 - > équipe en confiance et responsabilisée
 - Le client est impliqué : Scrum est conçue pour s'adapter rapidement aux évolutions et aux demandes du clients
 - > amélioration de la satisfaction client et élimination des "déchets" (fonctionnalités sans valeur ajoutée pour l'utilisateur)
 - La méthode se concentre sur la valeur métier des fonctionnalités et priorise l'implémentation en fonction de cela
 - > adaptabilité & réactivité
 - La méthode favorise des cycles de développement courts
 - > retour sur investissement rapide, pas d'effet tunnel

Phases d'une version de projet



- Le projet est découpé en versions (releases), elles-même divisées en itérations (appelées Sprint)



Pré-jeu

- Planning
 - Définition du backlog, des jalons, de l'équipe
 - Identification des facteurs de risques
 - Choix des outils & infrastructure + logistique
 - Budget, chiffrage & estimation du coût
- Architecture générale, conception générale

Jeu

- Sprints
- Développement (analyse, conception, codage) > livraison > révision > adaptation (~ Plan/Do/Check/Act)

Post-jeu

- Packaging global de la version
- Intégration & jeux de tests
- Documentation & manuels utilisateur globaux, accompagnement au changement

Rôles



PO

Product Owner

Représentant du client et des utilisateurs.
Formalise et priorise le backlog du produit



SM

Scrum Master

Meneur d'équipe, facilitateur, travaillant de façon rapprochée avec le Product Owner et l'équipe



E

Équipe

+/- 7 personnes
Elle doit tout faire pour délivrer le produit (organisation, profils, ...). La responsabilité est collective.



I

Intervenants ("chickens & pigs")

Observent & conseillent

Artéfacts (éléments produits)



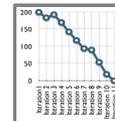
Backlog produit

Liste des macro-fonctionnalités de l'application, priorisées par leur valeur métier



Backlog de Sprint

Liste des tâches à implémenter sur un Sprint, classées par importance et état



Burn-down chart

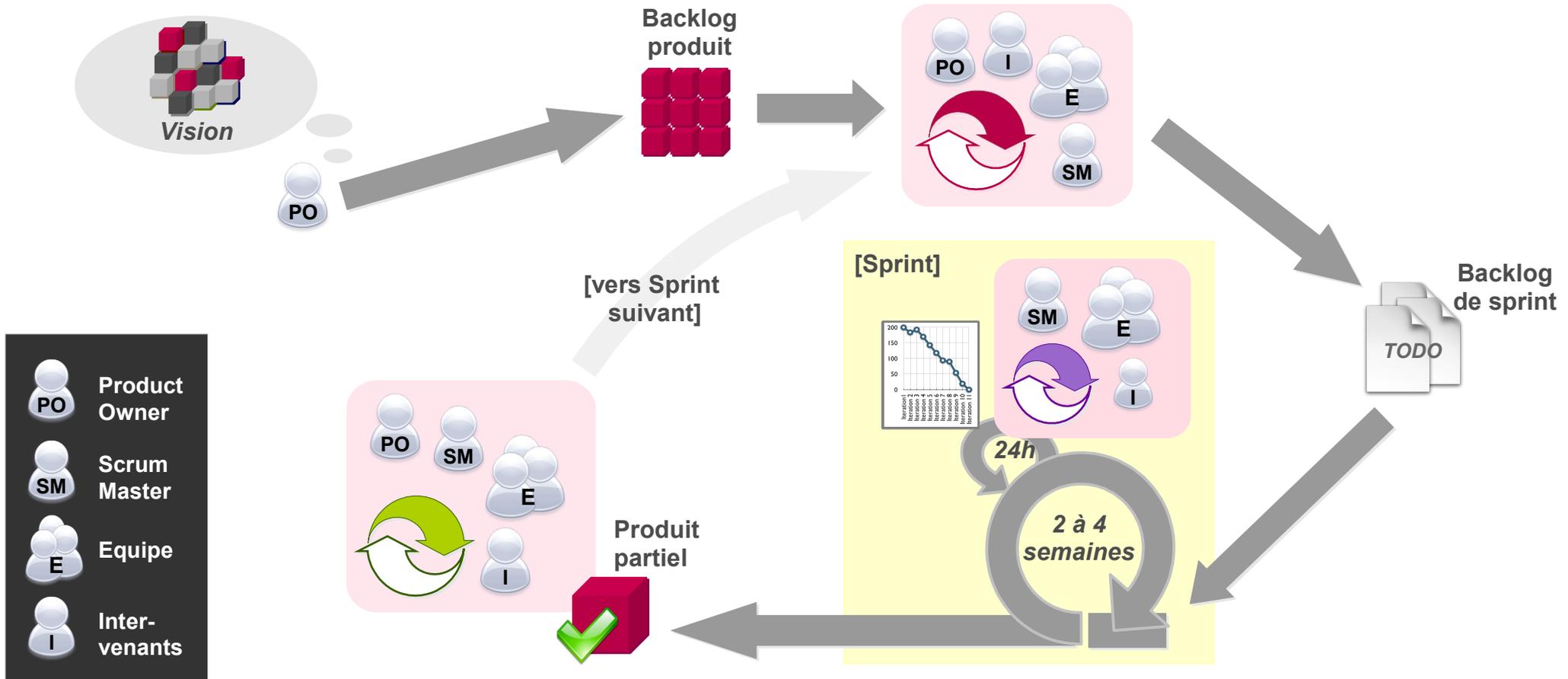
Diagramme permettant de suivre l'avancement du projet (représentation du Reste à Faire)



Produit partiel

Résultat du Sprint, testé et potentiellement livrable.

Le cycle de développement



-  **Product Owner**
-  **Scrum Master**
-  **Equipe**
-  **Intervenants**

Points de rencontres



Planification du sprint

Avant le démarrage du sprint, pour définir l'objectif et le plan détaillé de l'itération



Mêlée

Photo quotidienne du projet, pour identifier de nouvelles dépendances, répondre aux besoins de l'équipe, ajuster le plan de travail



Revue/Rétrospective de sprint

Revue pour vérifier la conformité de ce qui a été réalisé pendant le sprint
Rétrospective pour améliorer le sprint suivant

■ Activités & organisation de l'équipe

- Elle s'auto-organise, estime, fait les choix et se répartit les tâches pour délivrer le produit partiel
 - Intégration continue pour une optimisation des développements et des tests
 - Profils : développeurs, intégrateurs, experts (DBA, ...), ...
- Elle doit être dans un environnement optimisé pour atteindre l'objectif
 - Le Scrum Master veille au quotidien à ce que l'équipe soit au mieux
 - Les informations sont partagées : tableaux blancs, WIKI, ...
- Les tableaux de bord et indicateurs sont actualisés quotidiennement par le Scrum Master et permettent de suivre l'avancement de l'itération.

■ Méthodes

- L'estimation, l'analyse et la conception sont réalisées avec des méthodes agiles (cartes, index, croquis UML, ...)
- L'accent est mis sur les tests (Test Driven Development > mesure d'avancement par les tests, issu de l'XP)

■ Estimations :

- par analogie de préférence (utilisation du réalisé comme référence), ou par point de fonction
- Utilisation de méthodes agiles ("planning poker")

■ Planification

- Le planning doit permettre à l'équipe de "souffler" entre 2 sprints
 - Exemple : démos & rétrospective le vendredi matin pour le sprint n, planification du sprint n+1 le lundi matin ► vendredi AM libre pour de la veille, ...

■ Les estimations et la planification se basent sur le calcul de la vélocité

Vélocité estimée = (nb de jour/homme disponibles) x (coefficient d'effort)

réel, après application
congés, temps partiel, ...

- Le coefficient d'effort est une estimation sur la concentration de l'équipe. Il peut être évalué à partir des sprint précédents. Par exemple :

LAST SPRINT'S FOCUS FACTOR:

$$40\% = \frac{18 \text{ STORY POINTS}}{45 \text{ MAN-DAYS}}$$

THIS SPRINT'S ESTIMATED VELOCITY:

$$50 \text{ MAN-DAYS} \times 40\% = 20 \text{ STORY POINTS}$$

La correction d'anomalie peuvent être intégrées dans ce coefficient, les difficultés techniques peuvent être vues comme des "tech stories"

	Outils	Notes
Backlog (produit, sprint)	<ul style="list-style-type: none"> • Excel (cf. exemple http://blog.crisp.se/henrikkniberg) • Système de suivi de bug (Mantis, ...) • Spécifique Agile (VersionOne, Xplanner, ScrumWorks, IceScrum, Jira, Eclipse Plugin...) 	<p>Excel est suffisant pour la gestion de la liste, mais on ne peut se passer d'un outil de suivi des anomalies.</p> <p>Pas de best practices, si ce n'est faire au simple et au mieux pour l'équipe</p>
Communication & gestion documentaire	<ul style="list-style-type: none"> • WIKI • Tableau de tâches ("Sprint wall" + post-it, appareil photo...) • Espace de stockage 	<p>Imprimer les cartes du backlog pour les afficher au regard de tous</p>
Gestion de conf. & intégration continue	<ul style="list-style-type: none"> • Subversion • Hudson, Continuum ... 	
Pilotage (tableaux de bord)	<ul style="list-style-type: none"> • Indicateurs : RAF, vélocité, répartition de charge codage/test/correctif, ... <ul style="list-style-type: none"> • excel, • Spécifique Agile (cf. backlog) 	<p>A homogénéiser avec la gestion des backlogs</p> <p>Complexité à définir les diagrammes sous excel?? (burn-down chart...)</p>
Pilotage (suivis)	<ul style="list-style-type: none"> • Actions : idem > Excel ou spécifique • Risques : idem 	<p>A homogénéiser avec la gestion des backlogs</p>

- **Problématique : comment gérer un projet Scrum sur le plan contractuel, sachant que les concepts Agiles préfèrent la collaboration à la négociation de contrat?**
 - Contrats pour les forfaits "classiques"
 - prix fixe, périmètre fixe
 - 4 axes caractérisés et figés : Périmètre, budget, délai, qualité
- **Dans un contrat plus agile, on peut imaginer**
 - De la part du soumissionnaire, un engagement sur les moyens (compétences, méthodes, pilotage, ...) pour assurer la qualité
 - Estimation du rapport qualité/prix plutôt que prix le plus bas
 - Exemples de critères : niveau de qualité (méthodes, ...), fréquence des feedbacks, expériences des ressources, capacité à s'adapter, communication, maturité de la relation client
 - Un budget et des dates de livraisons (durée du sprint) fixés par le client
 - Un périmètre initial tracé de façon collaborative (backlog produit initialisé conjointement)

■ Différentes approches possibles pour l'agilité

- Contrat à engagement de moyens (~régie) : périmètre variable et coût fixe
 - Risques principaux : retard de livraison, dé-responsabilisation du soumissionnaire
- Co-sourcing
 - Equipe mixte, sur le site du client
 - Visibilité, transfert de compétences intrinsèque, risques partagés (des ressources des deux parties sont impliquées)
- Forfait collaboratif
 - Le coût et le délai sont fixés par le client ► le périmètre est la variable d'ajustement
 - Le soumissionnaire se positionne sur sa capacité et son savoir-faire (engagement sur des critères de qualité)
 - Le contrat a donc bien un prix et une date fixés. La différence est dans la fréquence de livraison et dans l'ajustement du périmètre.
 - Forte implication des parties
- Forfait avec engagement mesurable
 - Définition de points de mesures concrets et représentatifs (par exemple : Définition d'unités d'œuvre, engagement sur un niveau de qualité, ...)

Préconisé
par Valtech

Préconisé
par Aubry

Préconisé par
timwi

Forces

Questions

Copilotage avec le client

- Propice à une confiance réciproque
- Transparence sur l'avancement (résultats rapides)
- Adaptabilité : le client peut modifier son besoin

- Le client est-il prêt à s'impliquer?
- Comment gérer le changement par rapport au mode classique? (organisation, comportements, contrat, ...)
- Comment éviter de retourner dans le mode classique au 1er point dur?

Equipe & responsabilité collective

- Plus de responsabilisation, plus de motivation
- Pas d'individualisme
- Amélioration permanente

- Capacité à travailler en équipe, à partager?
- Capacité à se responsabiliser, à s'impliquer (envie?)
- Capacité à faire l'effort d'être polyvalent? (alternance codage, tests, correctif)
- Comment gérer les ressources non adaptées à ce mode de fonctionnement?

Forces

Questions

Cadre de production, méthodologies

- Orienté productivité et qualité
 - pas de travail superflu
 - Environnement adapté (IC)
 - Orienté tests (TDD)
- Plus de souplesse et de créativité
- Recherche de la simplicité
- Pilotage au quotidien

- Framework minimaliste > risque d'hétérogénéité? (si mauvaise communication dans l'équipe...)
- Comment justifier et estimer le surcoût lié aux tests et à la mise en place de l'environnement d'IC?
- Pour un 1er Scrum, comment estimer la charge de pilotage? (accrue par rapport à forfait "classique" + coût des outils si spécifiques agile)