

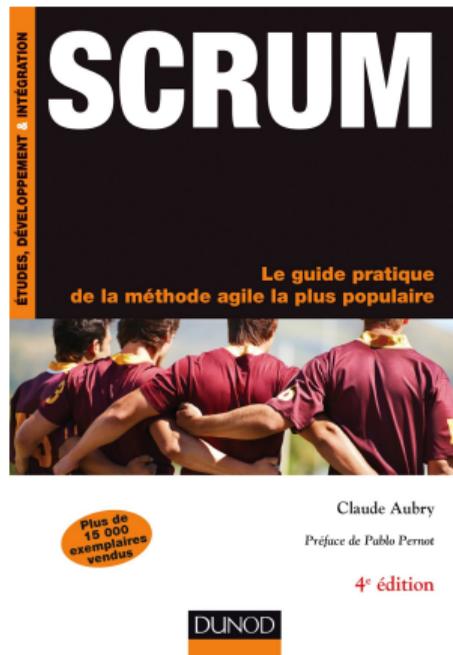
# Management agile - Scrum en pratique

Nuwan Herath

2022-2023

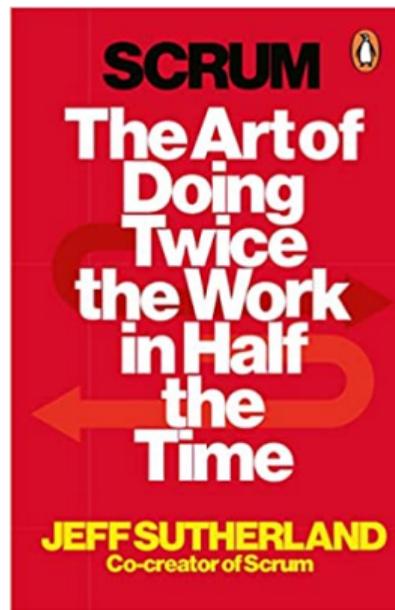
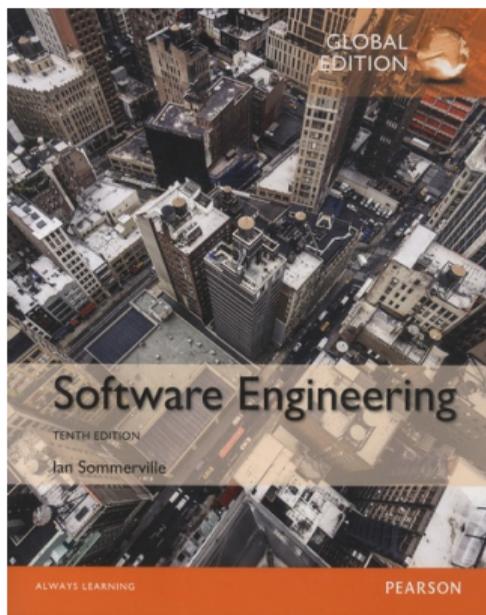
Ce cours se base en partie sur celui de Claude Godart

# Références



source des images

# Pour aller plus loin

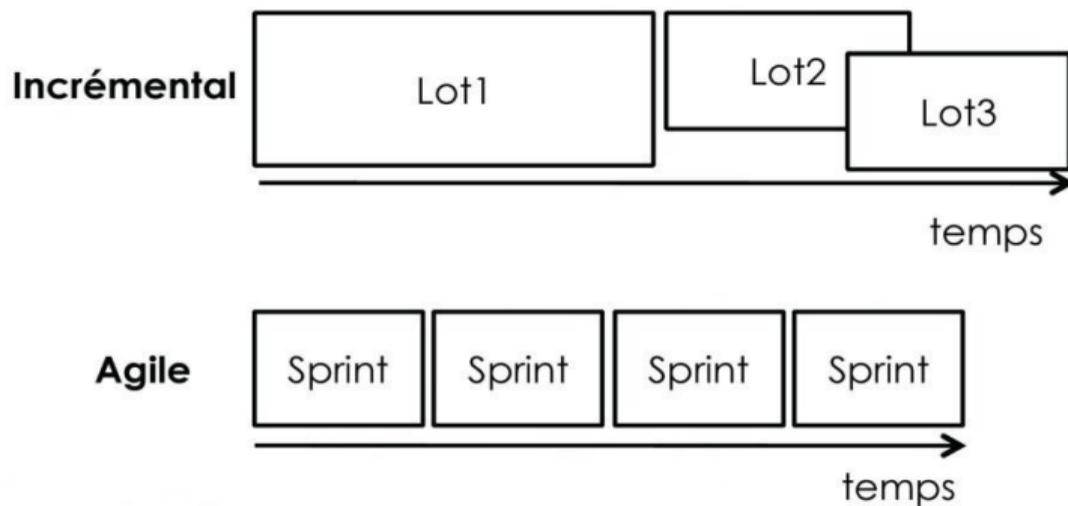


- 1 Le cœur de Scrum
- 2 Les acteurs de Scrum
- 3 Les objets de Scrum et plus. . .
- 4 Des indicateurs
- 5 Les événements de Scrum et plus. . .
- 6 Conclusion

# Le cœur de Scrum

# Un cycle agile

- Cycle itératif et incrémental
- Itérations courtes
- Séquence stricte
- Rythme régulier



# Le sprint

- Une équipe stable
- Une durée fixe (2-3 semaines voire 4)
- Planification de la participation des intervenants
- Budget fixe

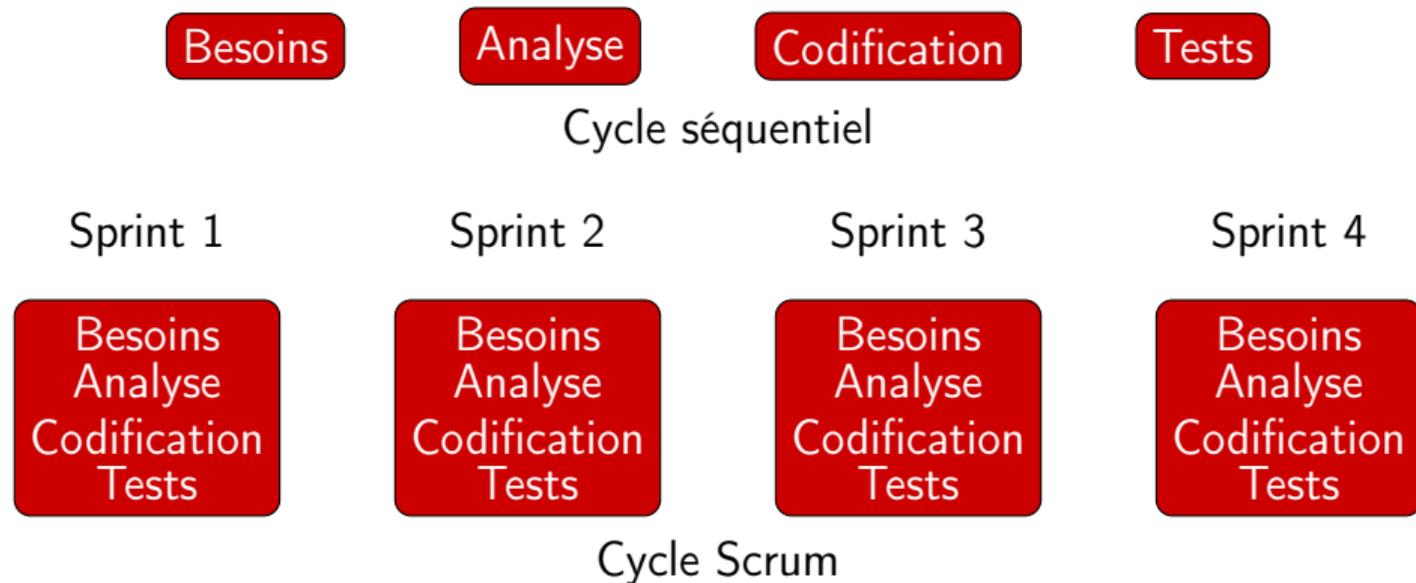
# La release

~~Une version : un incrément fonctionnel, une évolution technique~~

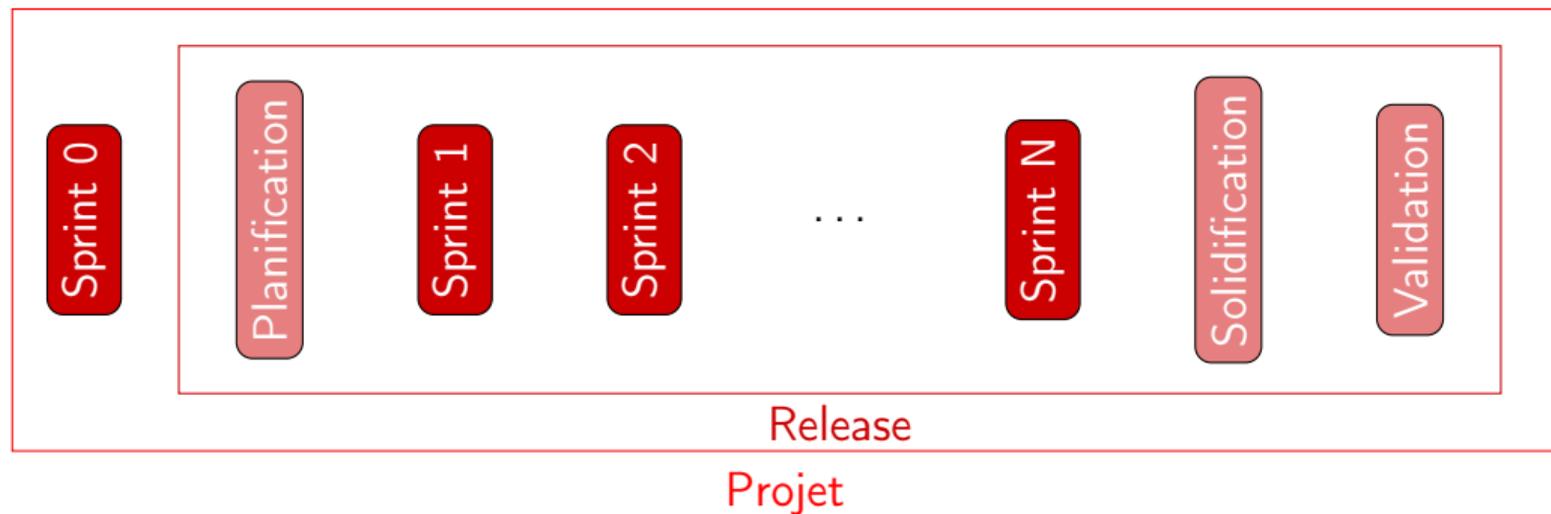
Livraison d'une feature (composée de features plus simples) réalisée pendant une période constituée de sprints (4 à 6, soit environ 3 mois)

Une nouvelle version peut être produite lors de n'importe quel sprint

# Le cycle de développement



# Les périodes de la release



# Le sprint 0

- Constitution de l'équipe, mise en place de l'environnement de développement
- Définition de la vision
- Production d'un backlog initial
- Première planification de la release

# Les acteurs de Scrum

# Les rôles

- Les parties prenantes (PP) : clients, utilisateurs, managers, experts, services et équipes impactés ...
- L'équipe Scrum
  - Le Product Owner (PO)
  - Le Scrum Master (SM)
  - Les développeurs (DEV)

# L'équipe Scrum

- Dans l'idéal 5 à 9 personnes (PO et SM inclus)
- Auto-organisée
- Pluridisciplinaire
- Stable

# Le Product Owner

Le Product Owner est responsable du résultat auprès des parties prenantes

Son rôle est de

- partager la vision du produit
- affiner le produit
- planifier

Il doit

- être disponible
- participer aux événements Scrum
- être impliqué dans l'avancement

# Le Scrum Master

Le Scrum Master n'est pas un chef de projet  
Il doit

- aider l'équipe à appliquer Scrum
- éliminer les obstacles
- faire preuve de pragmatisme

## Les objets de Scrum et plus. . .

Avec Scrum, les spécifications émergent de la collaboration entre le Product Owner et les autres membres de l'équipe et des retours des parties prenantes

# Le backlog

Le backlog est une liste ordonnée des choses à faire par l'équipe  
Il est

- public
- ordonné
- vivant

# Les éléments du backlog du produit

- La **story** : morceau de fonctionnalité visible d'un utilisateur

## Exemple

Pouvoir régler l'inclinaison de son siège

- L'**epic** ou la story épique : l'équipe considère qu'elle est trop grosse

## Exemple

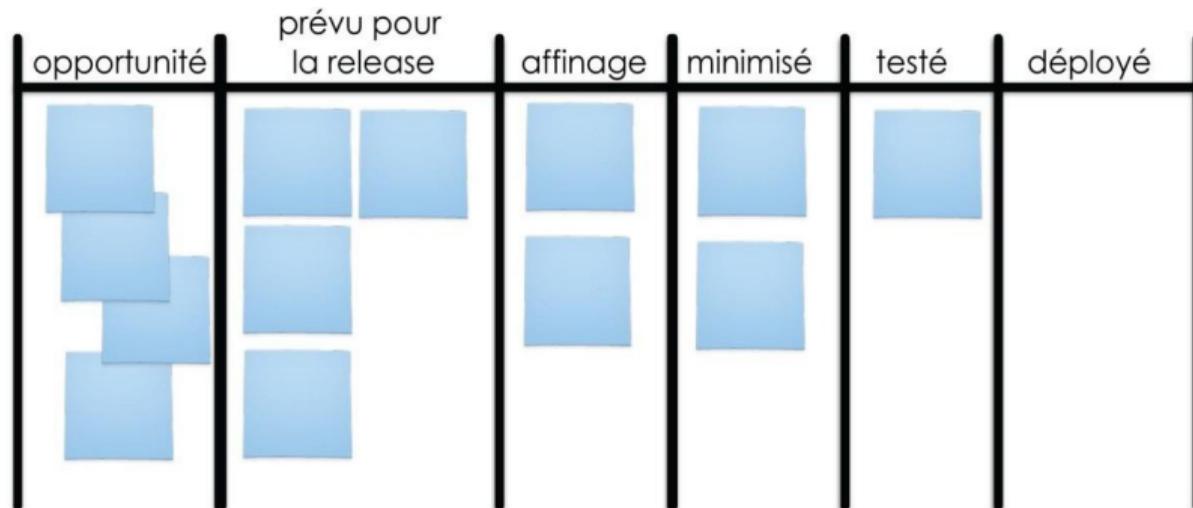
Pouvoir personnaliser sa voiture

- La **feature** : un service ou une fonction claire pour les parties prenantes

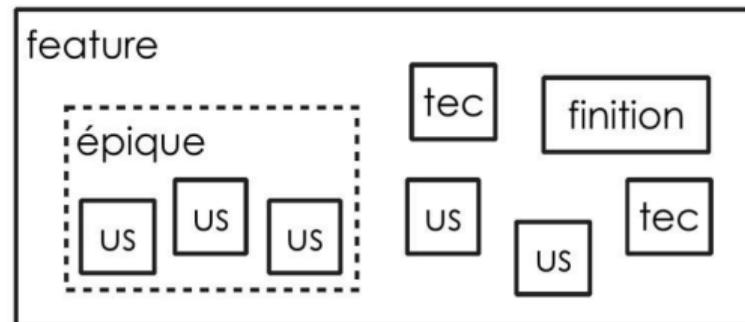
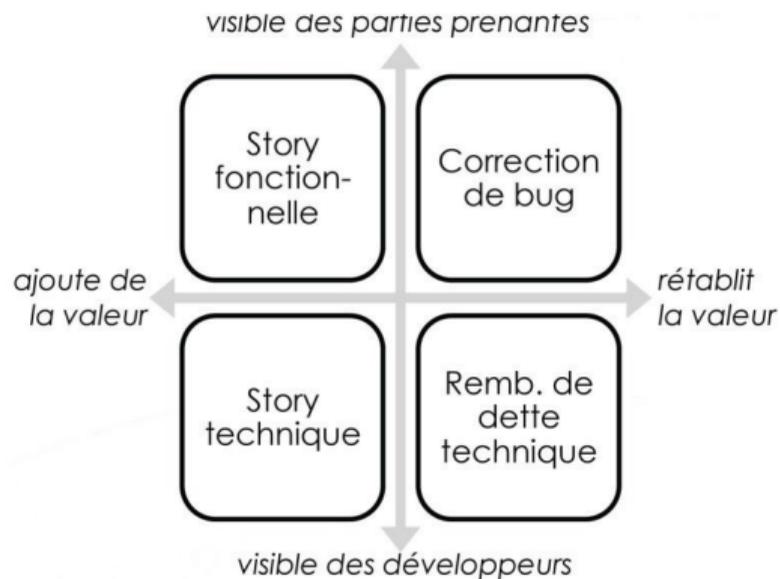
## Exemple

Rouler vers l'avant

# Le tableau de features



# Les grands types de story



# La vie d'une story

## Les 3 C

- Carte : l'histoire est courte et tient sur une carte (demi-page)
- Conversation : l'histoire est définie avec les gens du métier
- Confirmation : l'histoire est confirmée par des tests d'acceptation rédigés au même moment que celle-ci

La story devient prête pour pouvoir ensuite être finie

- Idée de story rédigée sur une **C**arte
- **C**onversation dirigée par le Product Owner qui inclut les gens du métier
- L'équipe apporte sa **C**onfirmation que la story est prête
- L'équipe réalise la story
- Le Product Owner apporte sa **C**onfirmation que la story est finie

## Description type d'une story

**En tant que** <acteur>, **je veux** <un but> [**afin de** <une justification>]

### Exemple

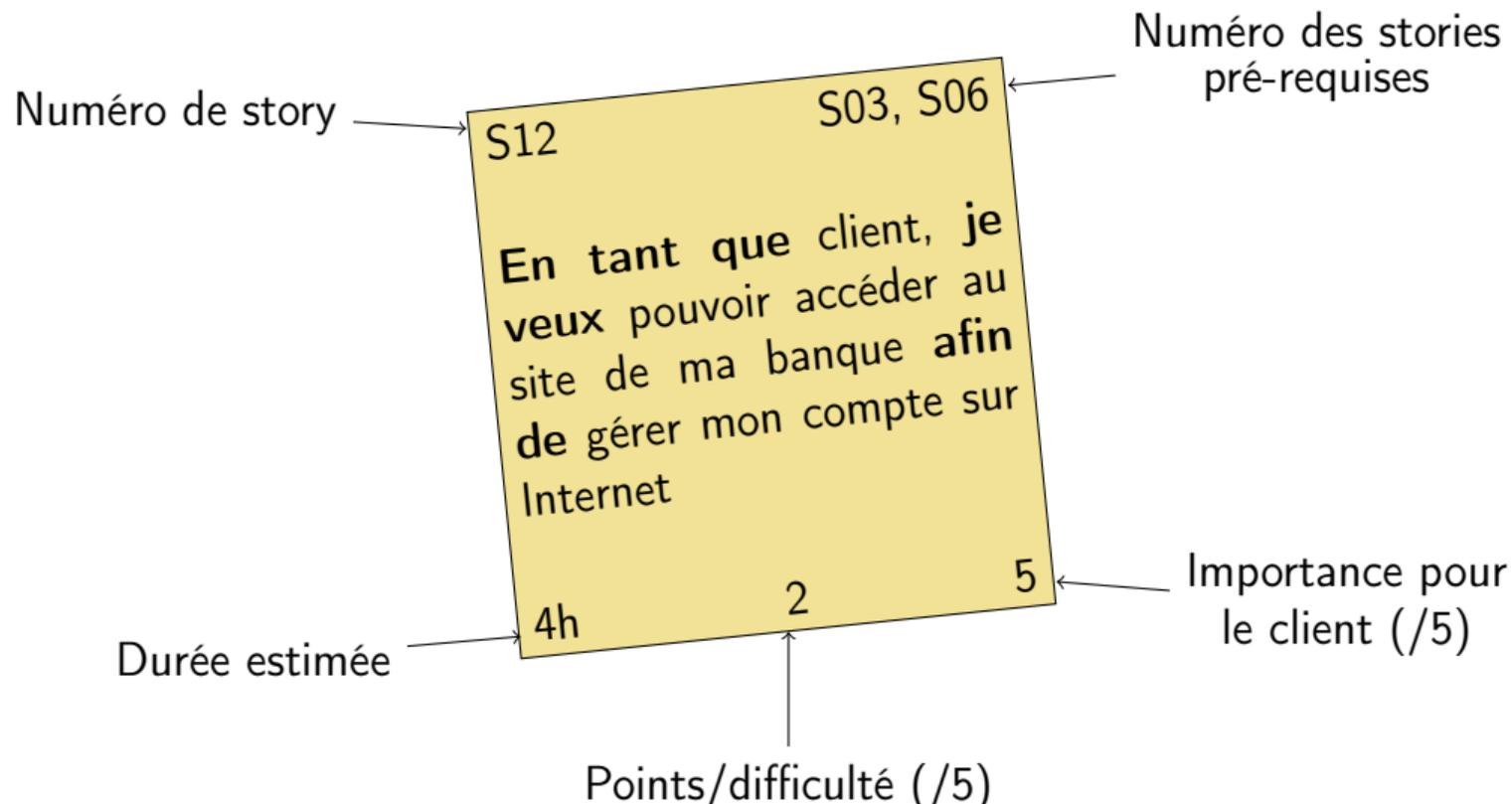
**En tant que** client, **je veux** pouvoir accéder au site de ma banque **afin de** gérer mon compte sur Internet

Une condition d'acceptation (voire plusieurs) :

**Etant donné** <le contexte> **quand je** <événement> **alors** <résultat>

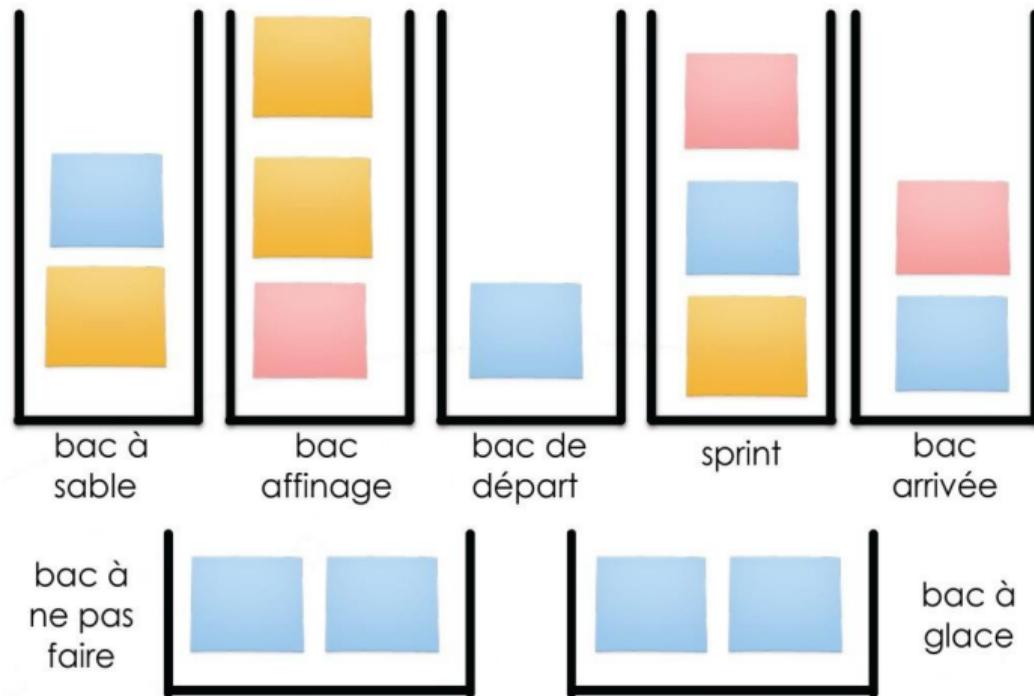
### Exemple

**Etant donné que** je suis sur la page de connexion et que j'ai entré un login et un mot de passe dans le formulaire et que le login et le mot de passe correspondent à un utilisateur enregistré, **quand je** clique sur le bouton "Se connecter" **alors** j'arrive sur la page d'accueil du site

Post-it<sup>®</sup> de story

# Le backlog

Une vue d'ensemble



# Le backlog

## L'affinage

Un travail d'équipe

Principalement le travail du Product Owner



- Approvisionner
- Estimer
- Ordonner
  - valeur
  - taille
  - dépendances avec des stories, des gens, une échéance ...
- Purger

# L'affinage lors du sprint 0

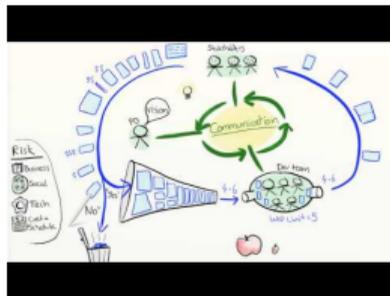
Avec les parties prenantes, on alimente le tableau des features

Puis

- on approvisionne le bac d'affinage,
- on ordonne, décompose et estime,
- on approvisionne le bac de départ pour le sprint 1

# Une vidéo

## Agile Product Ownership in a Nutshell, Henrik Kniberg



<https://www.youtube.com/watch?v=502ILHjX9EE>

# L'estimation des stories

Il faut estimer les stories ajoutées ou décomposées depuis la dernière séance d'affinage

# Le planning poker

## Déroulement

- 1 Les participants s'installent autour d'une table, placés de façon que tout le monde puisse se voir.
- 2 Le responsable de produit explique à l'équipe un scénario utilisateur (user story).
- 3 Les participants posent des questions au responsable de produit, discutent du périmètre du scénario, évoquent les conditions de satisfaction qui permettront de le considérer comme "terminé".
- 4 Chacun des participants évalue l'effort de développement de ce scénario, choisit la carte qui correspond à son estimation et la dépose, face vers le bas, sur la table devant lui.
- 5 Au signal du facilitateur, les cartes sont retournées en même temps.
- 6 S'il n'y a pas unanimité, la discussion reprend.
- 7 On répète le processus d'estimation jusqu'à l'obtention de l'unanimité.

Une procédure optimisée consiste, après la première "donne", à demander aux deux acteurs ayant produit les évaluations extrêmes d'expliquer leurs points de vue respectifs. Ces explications achevées et comprises de tous, une nouvelle estimation est produite et c'est alors la moyenne arithmétique de ces estimations qui est prise en compte.

(Wikipédia)

# Le planning poker

## Ressources

Vidéo : Le Planning Poker et les ESTIMATIONS Agile, Scrum Life



<https://www.youtube.com/watch?v=NZxcqei5qIE>

Outil en ligne : <https://planningpokeronline.com>

# La définition de fini

Qu'est-ce que finir ?

- Quand aurez-vous fini de coder ?
- Est-ce que tu as fini de ranger ta chambre ?

# La définition de fini

## Contenu

Lors d'un sprint on a un objectif de sprint qui concerne l'incrément du produit

La définition de fini consiste en une liste de vérifications

- sur la story,
- sur la feature
- et sur le sprint

## Exemple

Story	Respecter le standard de codage, passer les tests unitaires
Feature	Documentation marketing, export au format PDF

# La définition de prêt

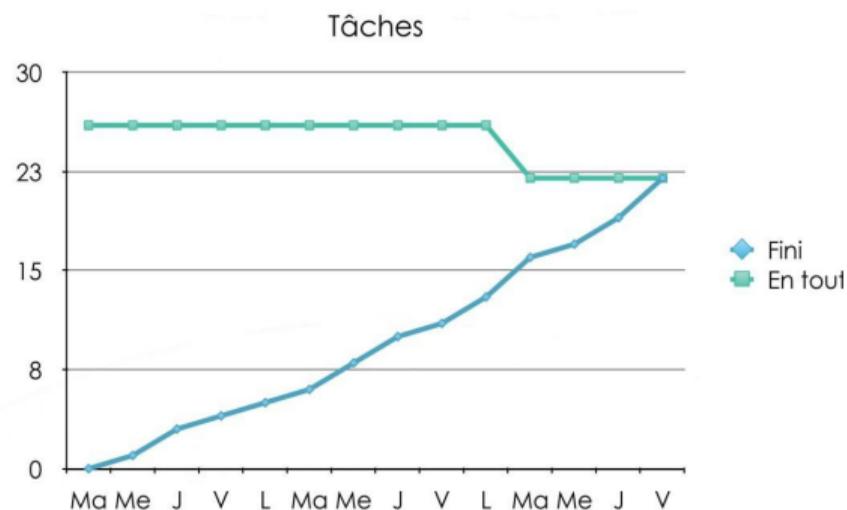
## Vérification des 6D pour une story

- Décomposée : ce n'est plus une story épique
- Débattue : elle a été discutée en équipe lors des séances d'affinage, au cours de conversations.
- "Dérivée" : les risques sur cette story sont réduits
- Définition de fini : on sait ce qui permettra de dire qu'elle est finie
- Démontrable : on saura la montrer à la revue
- Désirable : elle apporte de la valeur à quelqu'un

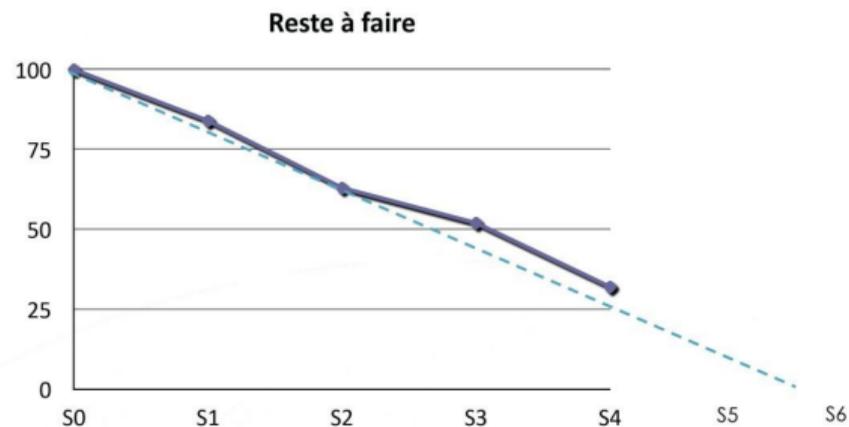
# Des indicateurs

# Des indicateurs pour le sprint et la release

## Burnup



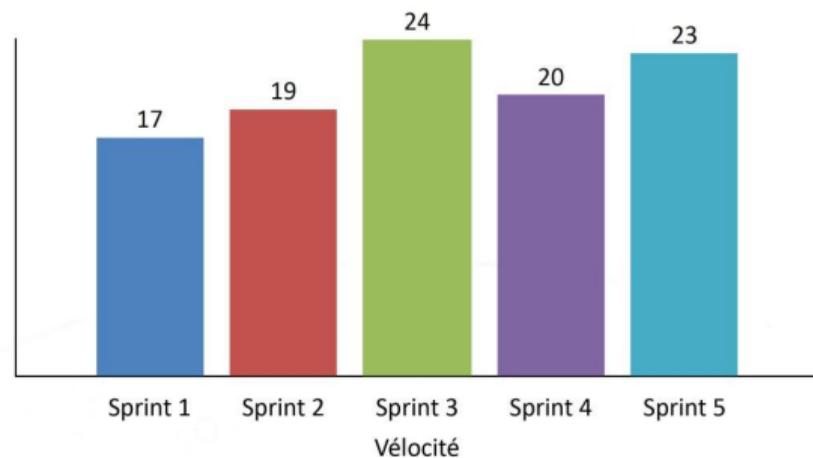
## Burndown



# Un indicateur pour l'équipe

La vélocité : nombre de stories finies par sprint

- Estime la capacité de l'équipe en nombre de points de stories par sprint
- Utilisée pour la planification de la release
- Affinée à la fin de chaque sprint
- Tendance à la stabilité



## Les événements de Scrum et plus. . .

# L'idéation

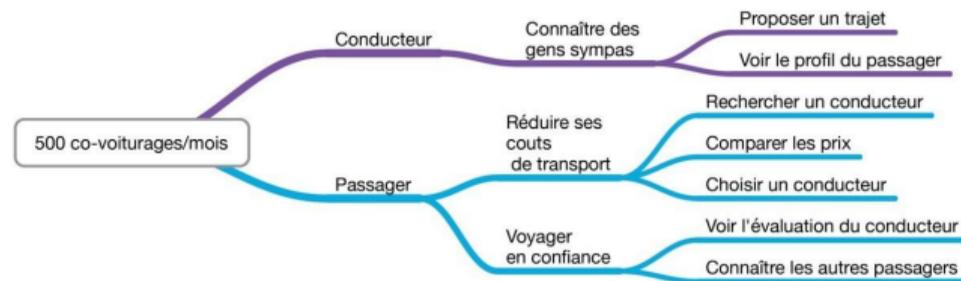
Naissance d'un projet : discussion entre client, manager et Product Owner

- Définition d'une vision commune
  - Identification des features → impact mapping
  - Identification des parties prenantes → acteurs
- Création d'un backlog de haut niveau du produit
  - Tableau ordonné des features
  - (éventuellement story mapping haut niveau)

# L'impact mapping

La carte des impacts se présente sous forme de carte heuristique (mindmap) qui contient, pour le produit envisagé :

- son objectif, au centre de la carte
- les acteurs au premier niveau
- les impacts attendus, au deuxième niveau
- enfin, les features qui sont la solution à ces impacts



Une carte qui donne la vision pour un site de co-voiturage

# La planification de la release

En quoi consiste-t-elle ?

- Affiner les risques, les incertitudes en fonction des retours de la revue et de la rétrospective du dernier sprint
- Ajuster la vélocité de l'équipe
- Affiner, (re)planifier le(s) prochain(s) sprint(s)
- Définition de prêt et fini
- Affiner, (re)planifier la future release

# La planification de la release

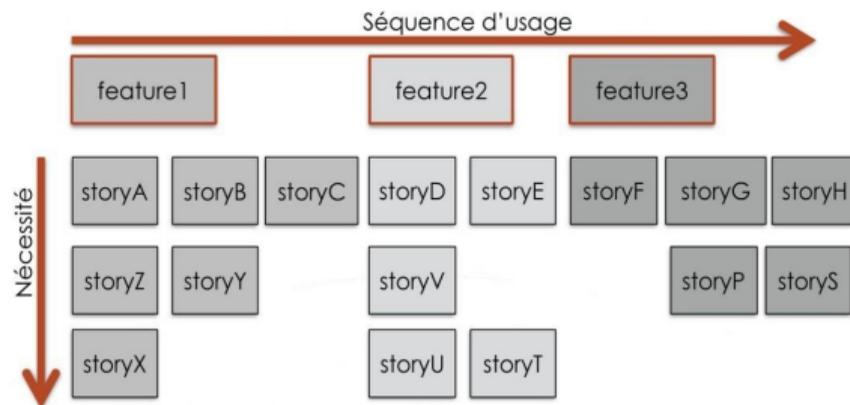
Faut-il un événement dédié ?

Excepté la première fois, il n'y a pas de réunion dédiée à la planification de release  
Beaucoup d'activités y contribuent

- lors de l'affinage : décomposer, ordonner, estimer, purger
- revue de sprint : mesurer la vélocité, présenter le plan aux parties prenantes

# Le story mapping

- Ordonnancement des features
- Décomposition en stories
- Organisation des releases



# Créer une story map

## Un exemple



# Créer une story map

## Procédure

- 1 Lister les tâches (commence par un verbe)
- 2 Regrouper en activité
- 3 Ordonner en une séquence qui raconte une histoire
- 4 Explorer les tâches plus en profondeur
- 5 Segmenter en release

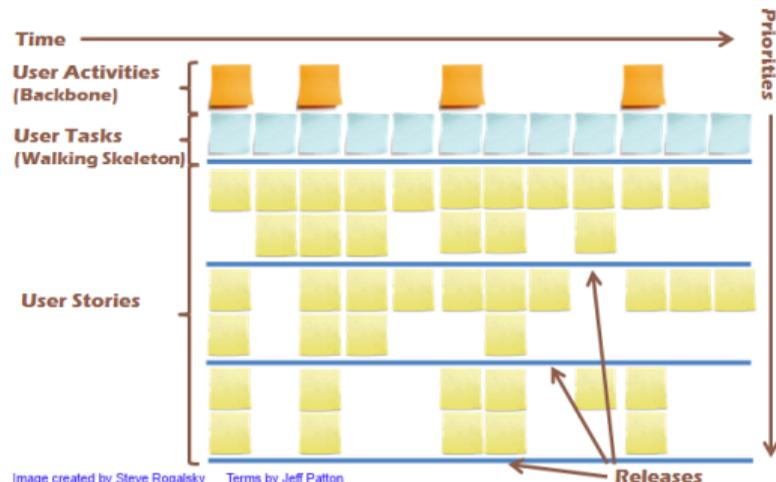
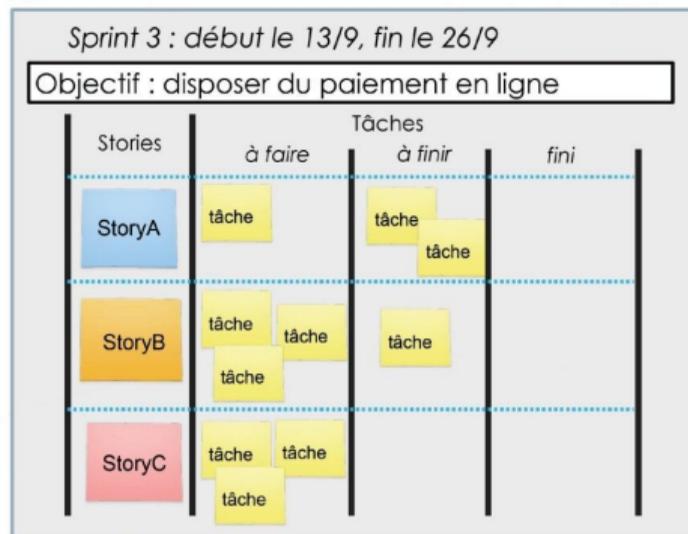


Image created by Steve Rogalsky <http://winnipegagilist.blogspot.com>  
 Terms by Jeff Patton <http://agileproductdesign.com>

Voir également [http://www.jpattonassociates.com/wp-content/uploads/2015/03/story\\_mapping.pdf](http://www.jpattonassociates.com/wp-content/uploads/2015/03/story_mapping.pdf)

# Le sprint



- Vérifier que les stories sont prêtes
- Les répartir au fur et à mesure par essaim
- Identifier les tâches
  - story 1 à 3 jours
  - tâche ~ 1 jour
- "Stop starting, start finishing"

# La mêlée

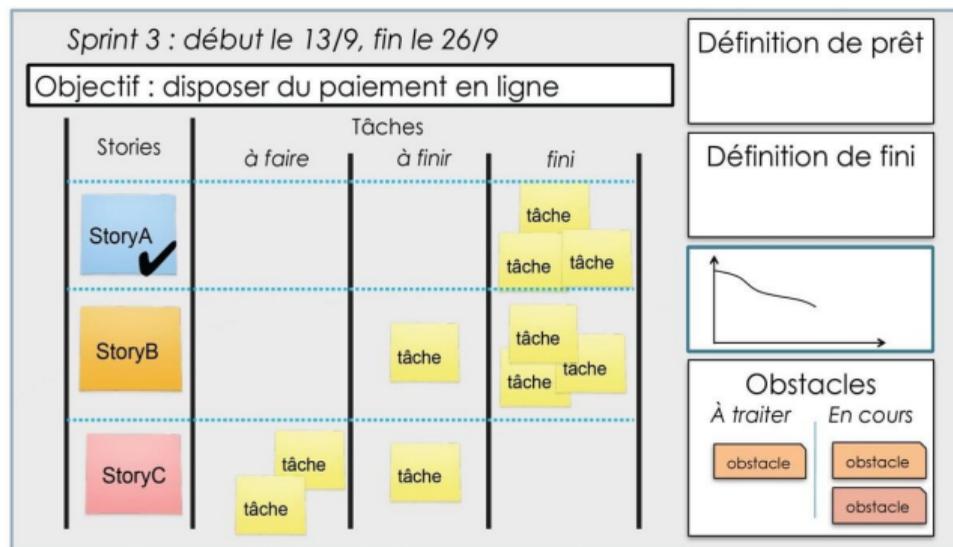
Daily Scrum Meeting / Stand up meeting / mêlée  
Inspection et adaptation au quotidien

**Durée** : 15 maximum

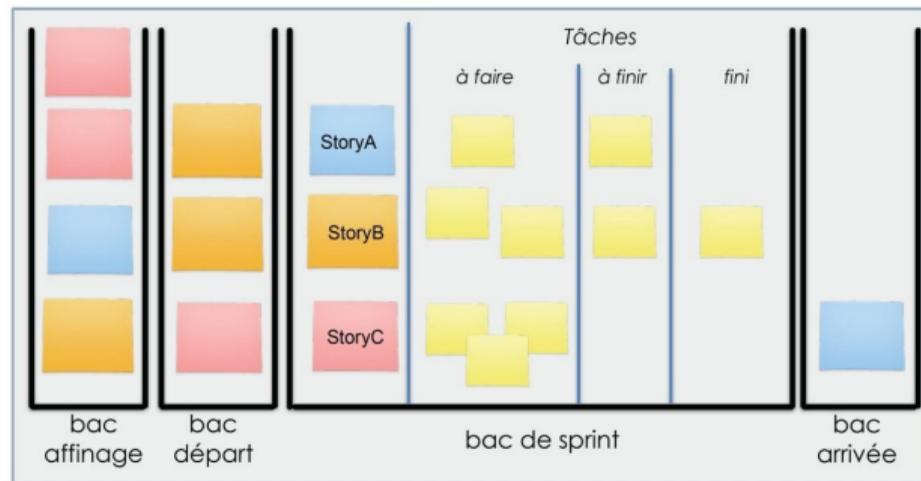
Chacun dit

- ce qu'il a fait la veille
- ce qu'il va faire dans la journée
- ce qui l'empêche d'avancer

# Evolution du tableau du sprint



# Un tableau à deux niveaux



# La revue de sprint

## L'un des douze principes du Manifeste agile

Un logiciel opérationnel est la principale mesure d'avancement.

Démonstration de l'incrément produit → visibilité et feedback  
On ne montre **que** ce qui est fini !

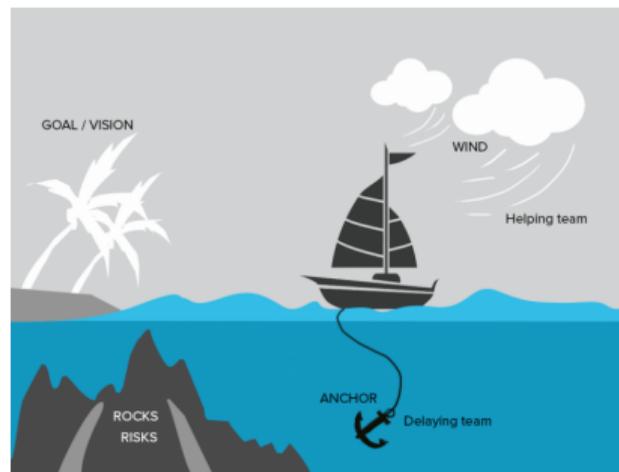
# La rétrospective

## Objectifs

- S'assurer que l'environnement est propice à l'expression de chacun
- Collecter les informations sur le sprint passé
- Identifier les idées d'amélioration
- Sélectionner les améliorations
- Adapter ses pratiques Scrum pour le prochain sprint

# La rétrospective

## Sailboat



<https://www.techagilist.com/agile/scrum/sailboat-or-speedboat-sprint-retrospective/>

# La rétrospective

## Starfish



<https://blog.myagilepartner.fr/index.php/2018/07/12/starfish-retrospective/>

# Solidification de la release

## Finalisation par l'équipe

- Tests de qualité de services (performances, coût, sécurité. . .)
- Documentation
- . . .

# Validation de la release

Validation par le Product Owner

Test d'usage avec le client et des utilisateurs

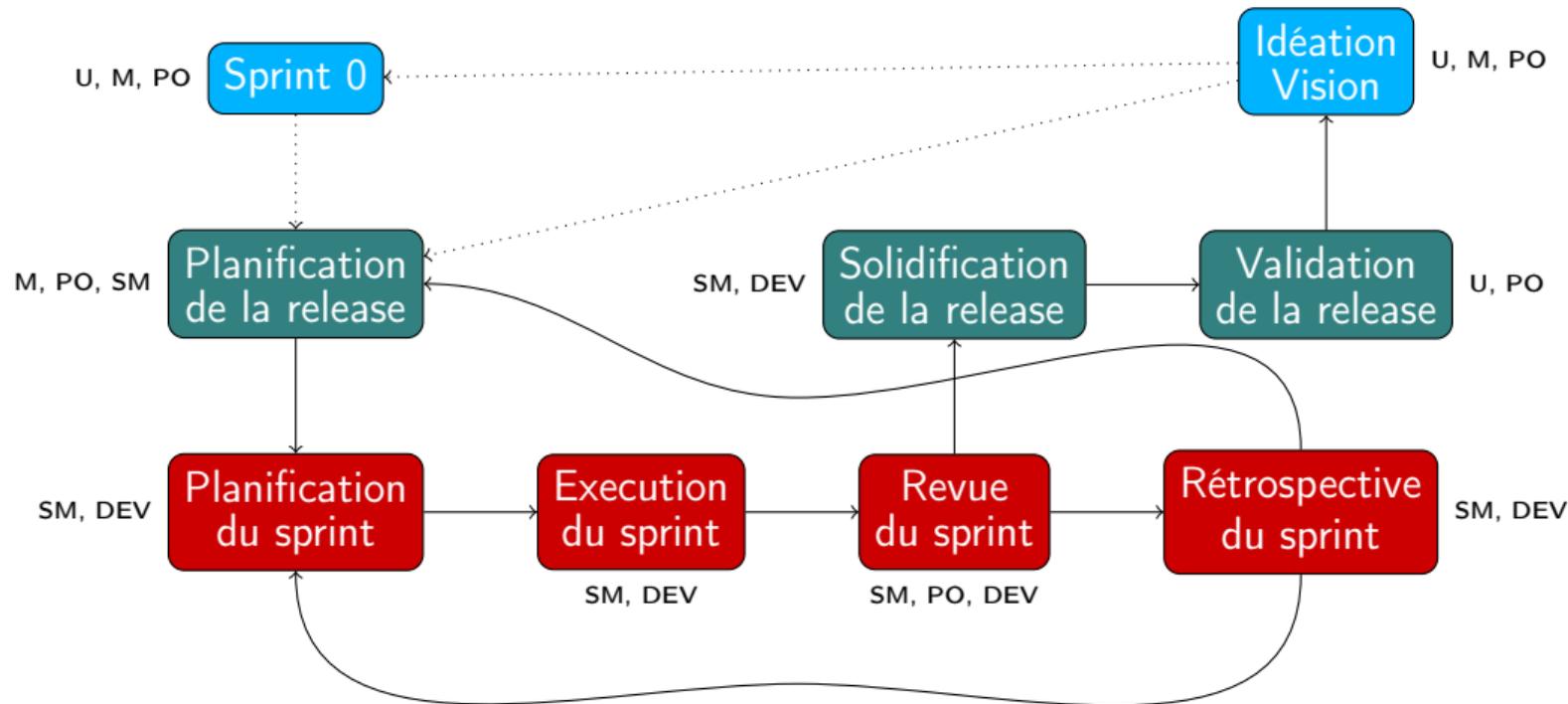
# Conclusion

# Notations

- Parties prenantes PP
  - Utilisateurs U
  - Manager M
  - (Experts EXP)
- Product Owner PO
- Scrum Master SM
- Développeurs DEV

# Le processus logiciel Scrum

Un diagramme approximatif



## Quelques outils

