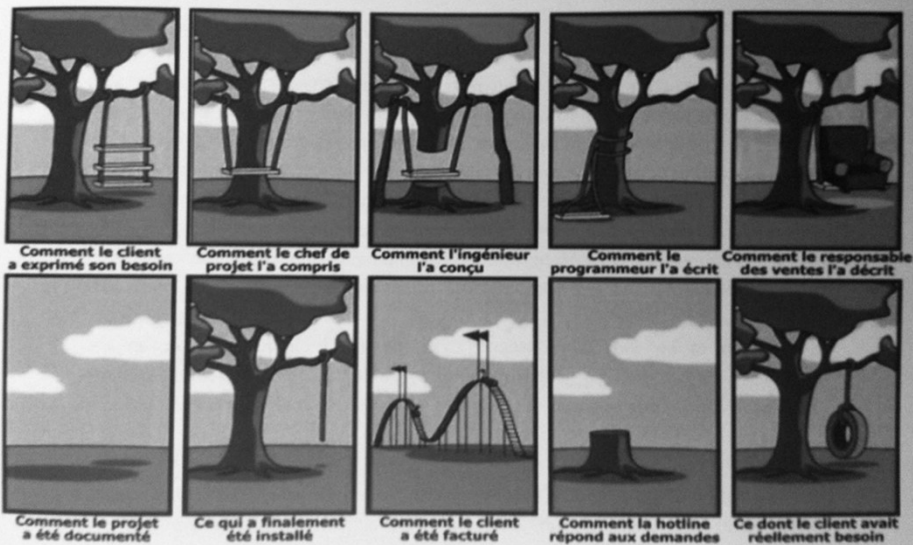


# Des méthodes Classiques vers Agile et Scrum



## Management Projet

L'enjeu principal du management de projet est de passer d'un objet à réaliser, abstrait, flou, compliqué, à un ensemble de constituants plus simples, plus précis et organisables. Il s'agit de mettre en œuvre une idée, un besoin de manière contrainte en respectant au mieux l'idée principale.



L'incontournable du chef de projet ; <http://www.anyideas.net>

⋮

## Cycle de Vie en Cascade

- Introduction en 1966
- Modélisation par Winston W Royce en 1970
- Il consiste à découper le projet en phases et de terminer chaque phase avant d'enchaîner.
- Chaque phase doit se terminer à une date précise et faire l'objet d'un livrable

⋮

⋮

## Cycle de Vie en Cascade

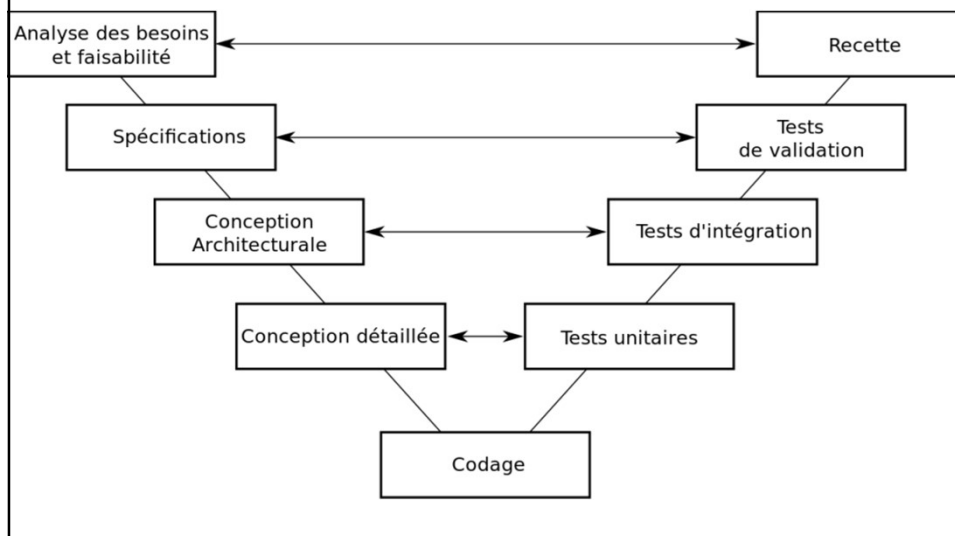
- Avantages :
  - Simplicité et intuitivité de mise en Œuvre
  - Cadrage et planification du processus de développement
  - Connaissance du planning par le client
  - Accent sur la documentation
  - Adapté aux projets stables et dont les spécifications évoluent peu

⋮

## Cycle de Vie en Cascade

- Inconvénients
  - Faible tolérance à l'erreur et au changement
  - Les défauts et les problèmes sont identifiés tardivement.
  - Les besoins doivent être déterminés dès le début du projet
  - Le client doit attendre la fin du projet. Effet Tunnel.
  - Forte dépendance au planning qui tient lieu d'indicateur qualité.

## Cycle en V



## Cycle en V

- C'est une évolution du cycle en cascade dans les années 80 et adaptés aux projets informatiques.
  - Qui n'ont pas une progression visible
  - Qui peuvent faire l'objet de nombreuses modifications dans les specs (pas de brick&mortar)
  - Dont le client maîtrise mal les aspects besoins/solutions

## Cycle en V

- Ce cycle permet de faire le lien entre les phases de conception ou de réalisation et les phases de tests.
- Les erreurs détectées lors des phases de tests remettent ainsi en cause la phase de conception associée.

## Cycle en V

- Avantages :
  - Simplicité de mise en œuvre
  - Connaissance du planning en avance par le client
  - Plus réactif que le modèle en cascade.

## Cycle en V

- Inconvénients
  - Faible tolérance à l'erreur et au changement
  - Difficulté de définir tous les besoins dès le début du projet
  - Défauts problèmes sont identifiés tardivement car les tests de validation sont effectués tardivement.
  - La validation par le client est tardive, le risque est élevé de rejets, les coûts de non qualité importants. Effet- Tunnel

## Cycle de vie en spirale

- Modèle introduit en 1988 par Barry Boehm
- S'appuie sur le modèle en V avec une étape de gestion des risques préalable à l'analyse des besoins.
- Il s'appuie sur la livraison régulière de prototype (IHM) à destination du client.

## Cycle de Vie en Spirale

- Avantages :
  - Modèle adaptatif
  - Livraison de prototypes ce qui permet une prise en main de l'utilisateur plus rapide (formation cycle d'utilisation, bac à sable)
  - Augmentation de la visibilité et du Feed Back client
  - Meilleure gestion des risques reportées en amont.

## Cycle de Vie en Spirale

- Inconvénients
  - Adapté aux gros projets (on peut fournir plusieurs prototypes)
  - Nécessité d'une méthode incrémentale stricte pour ne pas livrer une nouvelle fonctionnalité en omettant ou en modifiant un point qui fonctionnait précédemment.
  - Risque de dérive du cahier des charges au contact du client et donc des problèmes de facturation.

## Cycle Itératif

- A la suite du cycle en spirale, le cycle itératif propose de réduire le nombre de phases du projet.
- Chaque itération est constituée de la définition des spécifications, de l'étape de codage, la réalisation de tests, puis une évaluation du travail fourni permettent de faire le bilan sur les difficultés rencontrées et l'évolution des fonctionnalités.

•  
•  
•

## Cycle itératif (suite)

- Tous les livrables validés sont déployés vers l'utilisateur final (version beta, documentation...).
- On retrouve l'idée du cycle PDCA (Deming).

• • • • • • • • •

•  
•  
•

## Cycle itératif

- Avantages
  - Flexibilité et souplesse : prise en compte des retours et des difficultés rencontrées à la fin de chaque itération et possibilité de varier par rapport au besoin initialement défini.
  - Livraison de prototypes ce qui permet une prise en main rapide de l'utilisateur final (possibilité d'une démarche mkg stratégique).
  - Amélioration continue

• • • • • • • • •



## Cycle itératif

- Inconvénients
  - Orienté gros projets : les prototypes ne sont pas nécessaires si projet simple.
  - Nécessité d'être vigilant lors des phases de tests pour ne pas livrer une nouvelle version moins fonctionnelle que la précédente.

## Agir...



⋮

## Méthodes Agiles VS. Itératives

- L'agile se distingue de l'itératif car il est à la fois incrémental et itératif.
  - Quand une équipe agile effectue une itération, elle produit un incrément.
  - Chaque incrément va ajouter de nouvelles fonctions suite aux retours des utilisateurs finaux. Et il peut être affiné avec des fonctionnalités existantes, ce qui le rend itératif.

⋮

⋮

## Méthodes Agiles

- Du modèle itératif introduit par B. Boehm (1986) vont se détacher les méthodes Extreme Programming (XP), Scrum ou encore Crystal Clear.
- En 2001 aux USA un ensemble d'experts ont émis le « Manifeste Agile » qui tient lieu de fait fondateur des « Méthodes Agiles »

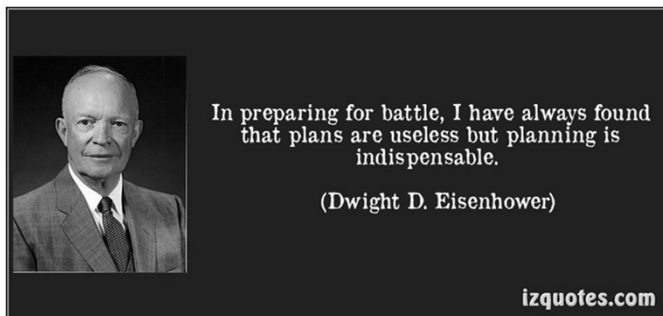
⋮

## Méthodes Agiles

- Manifeste Agile
  - Les individus et leurs interactions
    - .... Plutôt que des processus et des outils
  - Un logiciel Opérationnel
    - ....Plutôt qu'une documentation exhaustive
  - La collaboration avec le client
    - ....Plutôt que le contractualisation des relations
  - L'acceptation des changements
    - .... Plutôt que le suivi d'un plan

## Méthodes Agiles

- Le manifeste agile précise que les éléments de droite sont importants, mais moins que ceux de gauche.



...

## Conférence sur les méthodes Agiles

- En soirée 21 novembre 2017 UPS,  
organisée par Upsilon UPS.  
Spécialiste : Tibault Lesur

### 2001 - Le manifeste Agile :

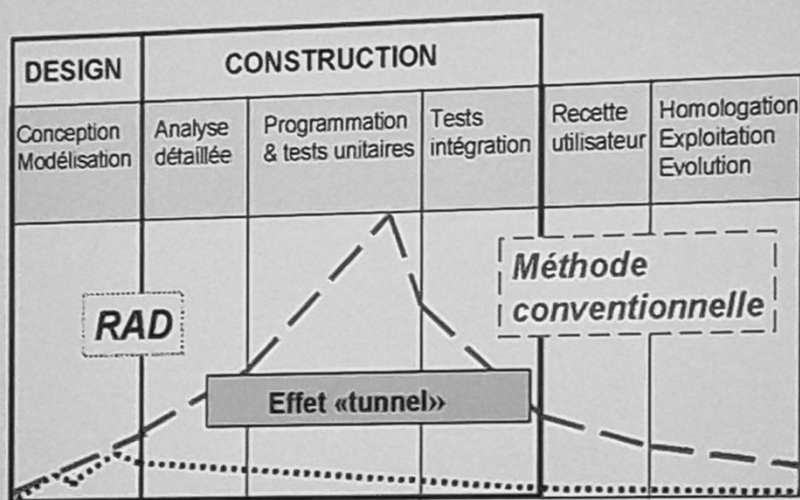
*Nous découvrons comment mieux développer des logiciels par la pratique et en aidant les autres à le faire.  
Ces expériences nous ont amenés à valoriser :*

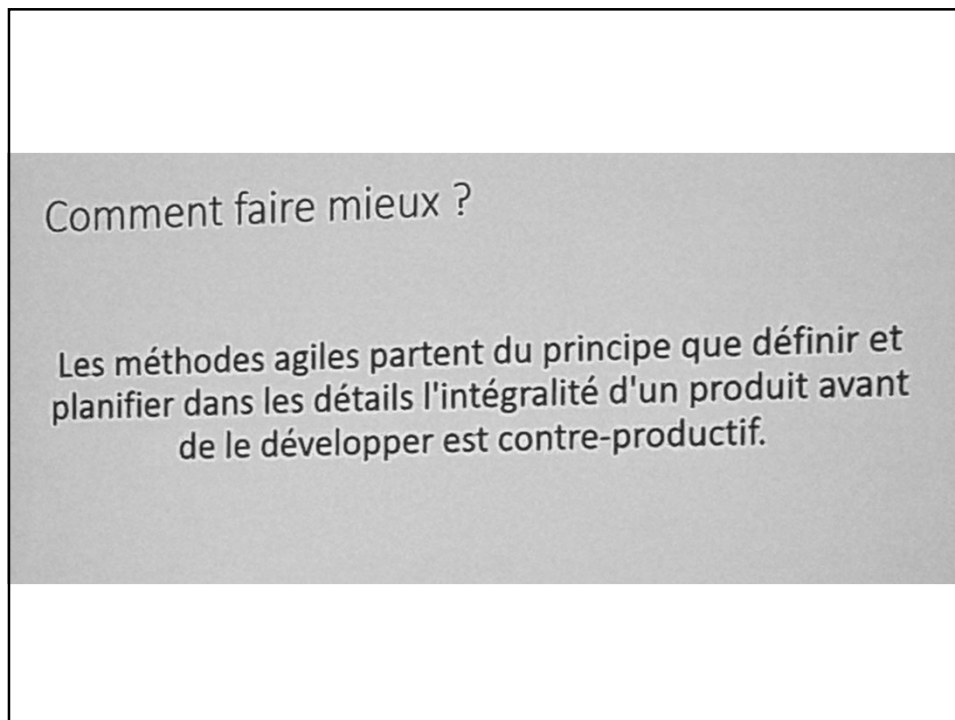
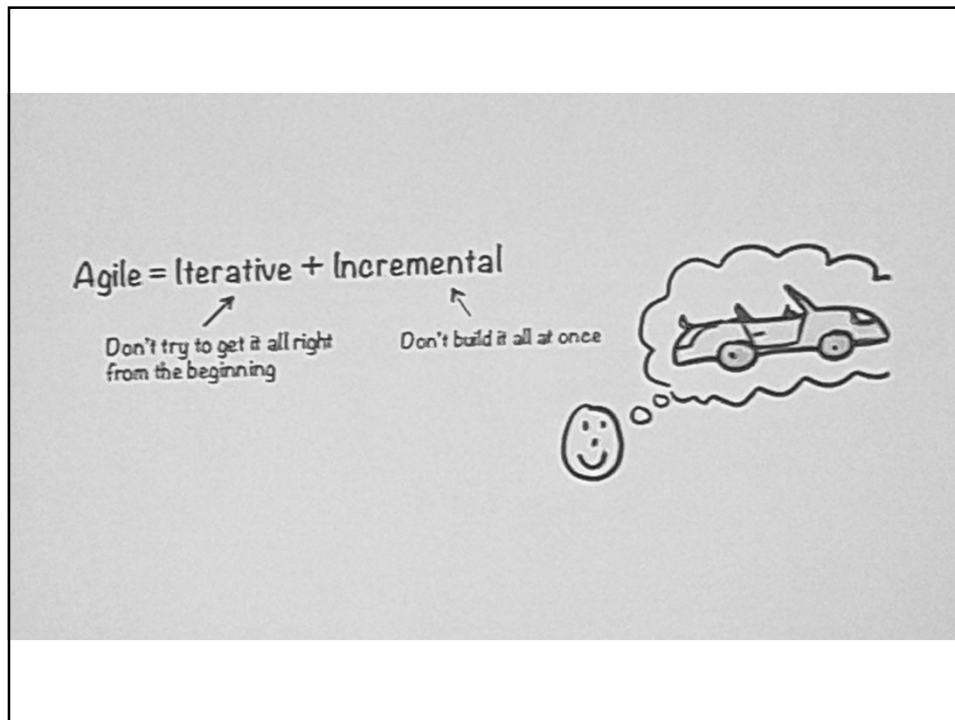
- ✓ *Les individus et leurs interactions* plus que les processus et les outils
- ✓ *Des logiciels opérationnels* plus qu'une documentation exhaustive
- ✓ *La collaboration avec les clients* plus que la négociation contractuelle
- ✓ *L'adaptation au changement* plus que le suivi d'un plan

*Nous reconnaissons la valeur des seconds éléments, mais privilégions les premiers.*

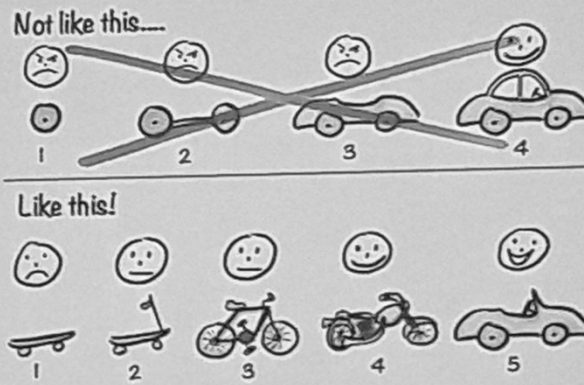
## Les apports de l'approche Agile face aux limites du cycle en V :

- **Briser l'effet tunnel** en assurant des livraisons régulières et incrémentales
- **Favoriser la collaboration** entre les membres de l'équipe (briser les sillots)
- **Etre « User Centric »** : mettre l'utilisateur au centre de la réflexion
- **Considérer l'inconnu** comme partie prenante de la vie !

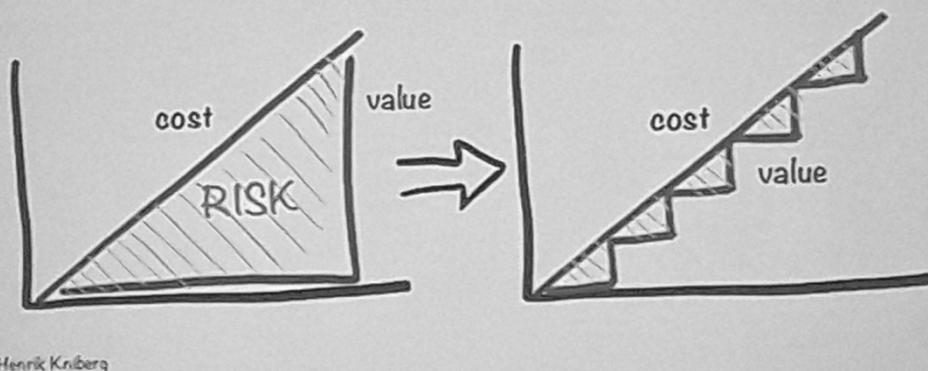




On opte pour approche incrémentale et itérative...

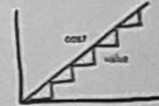


Une manière de réduire le risque :



## En concrètement... ca se passe comment ?

- Prenons un exemple d'équipe Scrum en tant que **cadre méthodologique**
- Un organisation d'équipe favorisant la collaboration :
  - De préférence pluridisciplinaire (ca dépend des cas)
  - De taille réduite (7 personnes max)
  - Le workspace adapté avec du management visuel
  - Horizontales et auto-organisées (personne n'est le chef de personne)
- Des itérations rythmées par des cérémonies et des feedbacks réguliers
- La délivrance de valeur à la fin de chaque itération



## Du management visuel

Active					Révisé			
Item	Projet	Assigné	En cours	Terminé	Statut	Titre	Durée	Commentaire
P1	P1.1	P1.2	P1.3	P1.4	En cours	P1.1	1h	
P2	P2.1	P2.2	P2.3	P2.4	En cours	P2.1	1h	
P3	P3.1	P3.2	P3.3	P3.4	En cours	P3.1	1h	





## La composition d'une équipe Scrum

- Le « **Product Owner** » qui porte la vision du produit à réaliser (représentant généralement le client).
- Le « **Scrum Master** » garant de l'application de la méthodologie Scrum.
- L'**équipe de développement** qui réalise le produit.
- **Les parties prenantes** : donnent des idées, feedback et expriment les besoins



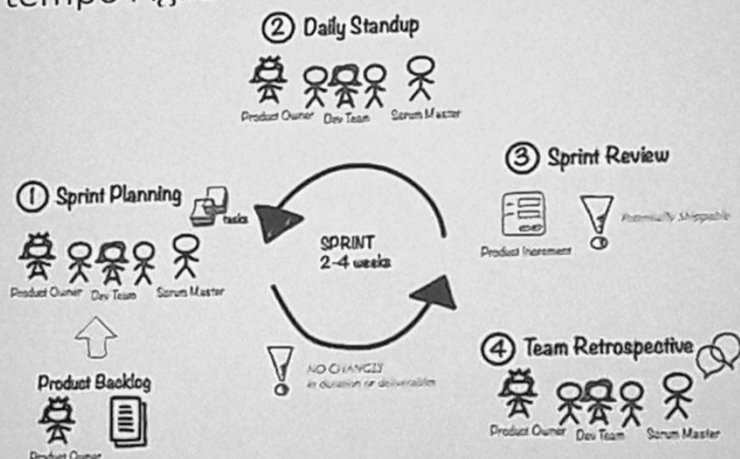
## Un peu de jargon...

- **User story** : fonctionnalité du produit assez précisément décrite pour être incluse dans un sprint
- **Tâche** : Ensemble des étapes nécessaires pour réaliser une User Story
- **Product backlog** : ensemble des User Stories nécessaires pour réaliser le produit
- **Vélocité** : C'est la capacité de l'équipe à délivrer de la valeur

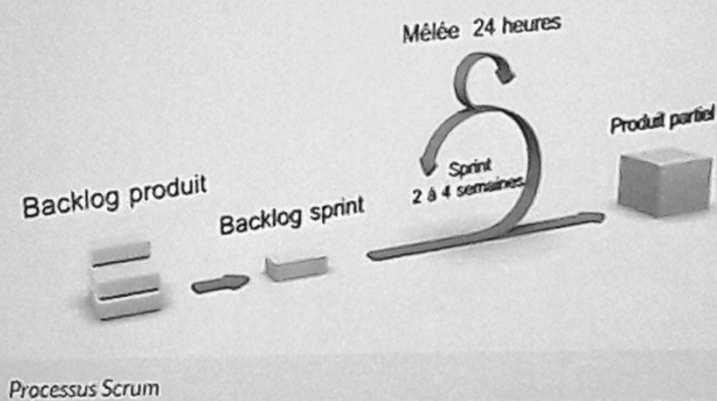
## ...et des cérémonies !

- **Sprint** : une itérations de 2 à 4 semaine max
- **Poker planning** : séance d'estimation des taches
- **Sprint planning** : cérémonie de définition de ce que l'on va inclure dans le sprint
- **Standup meeting** : mêlée quotidienne, debout 15min max
- **Point de déminage** : un créneau en cas de problème
- **Sprint review** : démo de ce qui a été réalisé au cours du sprint
- **Rétro**: on refait le match en équipe... pour s'améliorer !

## Le tempo Agile :



## Le tempo Agile autrement :



## Une définition :

« 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 des clients »

## Les valeurs véhiculées par l'agilité...

- **Communication** (se parler, s'écouter, jouer en équipe)
- **Courage** (se dire la vérité dans la difficulté, engagement)
- **Feedback** (faire des retours réguliers sur ce qui a été fait, s'y exposer)
- **Simplicité** (faire juste ce qu'il faut, des interfaces et process simples)
- **Respect** (des personnes, des valeurs, de soi-même)
- **Humilité** (ça se passe de commentaire)

A l'échelle d'une entreprise toute entière, pour certains l'Agilité c'est aussi ça :

