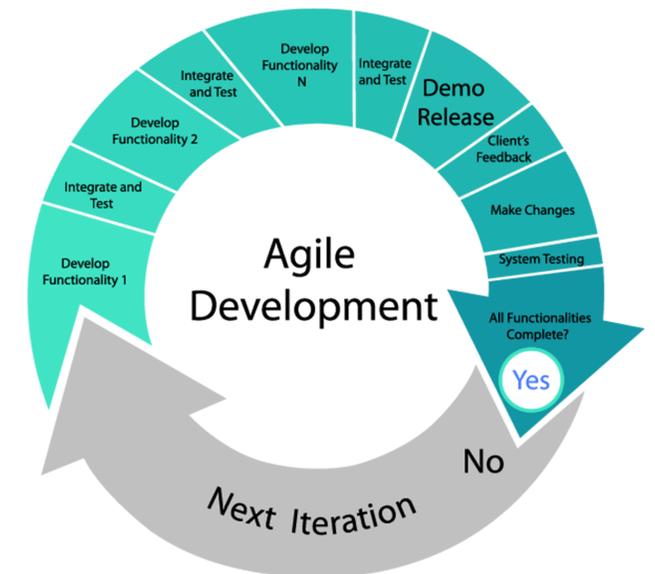
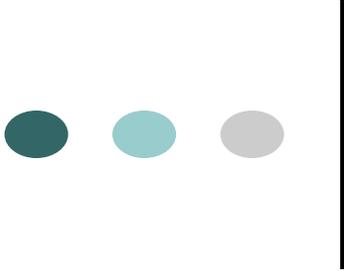


Introduction aux méthodes agiles



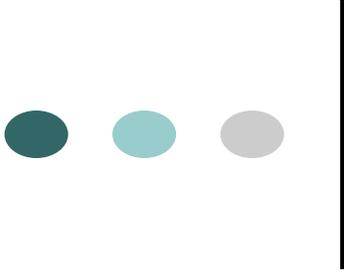
V. Deslandres ©
IUT de LYON
Parcours PEL - s4





Objectifs du module (16h)

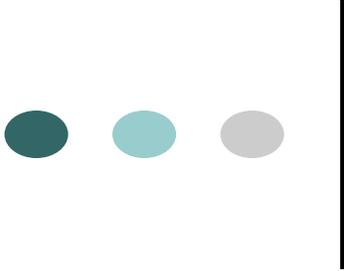
- Prendre connaissance des principes de l'agilité
- Découvrir les méthodes agiles
 - Scrum surtout
 - Point de vue Développeurs et peu Gestion de Projet
 - Les mettre en œuvre sur un mini-projet



Evaluation du module

- Backlog : les différentes versions (20%)
- Présentation du mini-projet (30%)
- QROC (fin de module, sur cours + vidéos, 50%)

QROC : *Question à Réponses Ouvertes et Courtes*



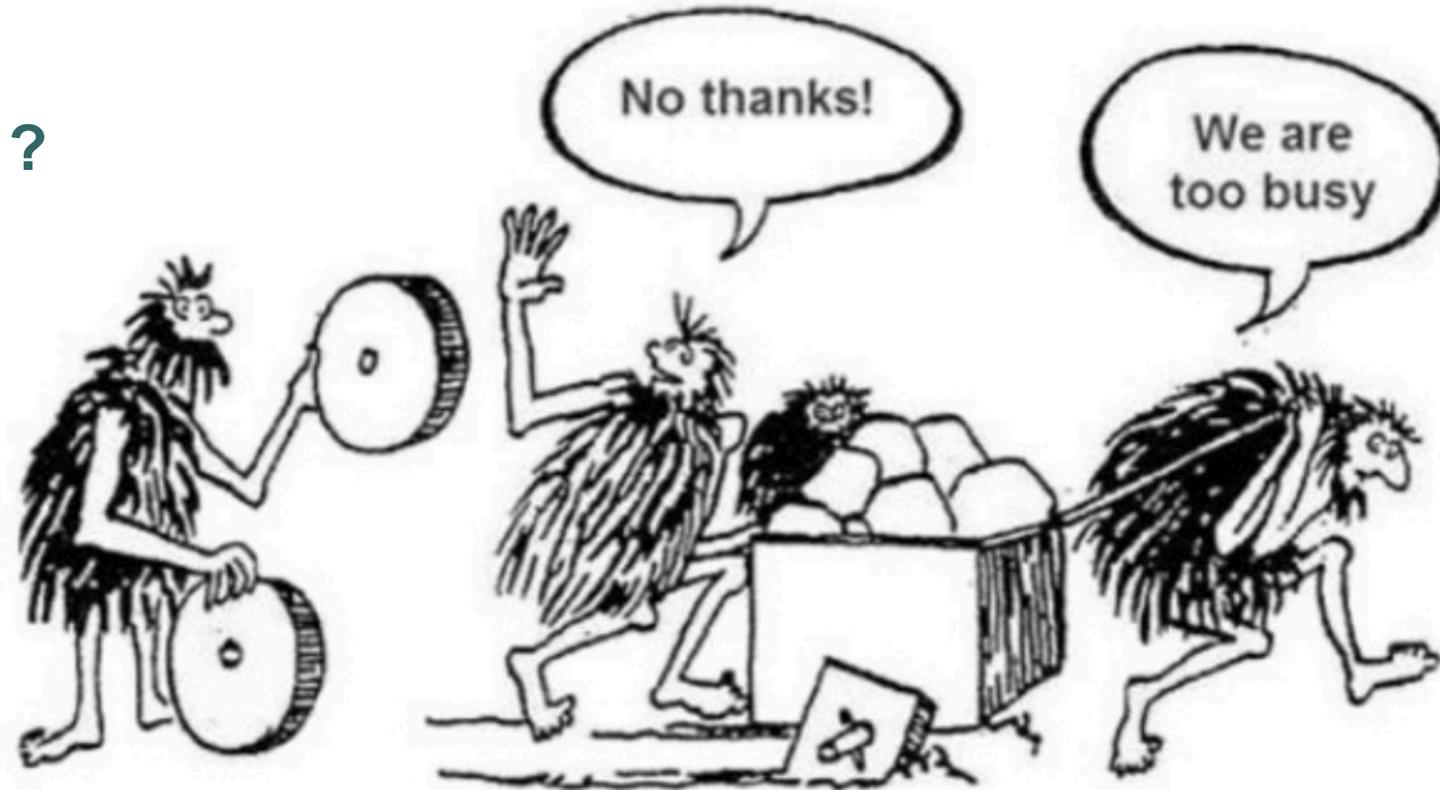
Plan de cette séance

- Introduction aux méthodes **agiles** (*ce petit support*)
- **Introduction à SCRUM**
 - La vision et le backlog Produit
- **Le mini-projet**
 - Présentation, planning

Pour vous c'est quoi, être agile ?

- CVDA en s2
- Conférence du professionnel M. Combier...

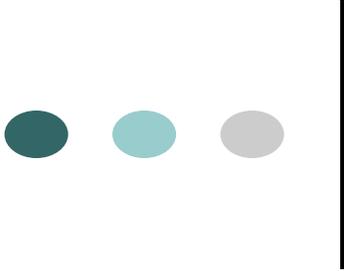
Un effet de mode ?



C'est quoi, être agile ?

- Plusieurs réponses possibles, suivant le prisme choisi, la plupart complémentaires :





Génie Logiciel : le constat

- **Années 2000**, un postulat : le cycle de développement traditionnel des logiciels ne correspond plus aux **contraintes** et aux **exigences** du marché.
- Entre le moment où le besoin est exprimé et le moment où le logiciel est délivré il peut se passer plusieurs mois.
- Les utilisateurs ne savent vraiment ce qu'ils veulent **qu'après** avoir vu une première version de l'application
- Agile : origine USA, en **2001**, 17 consultants, sur la production de logiciels, 20 ans déjà !

Faiblesses des méthodes classiques

Peu d'adaptation aux changements du client

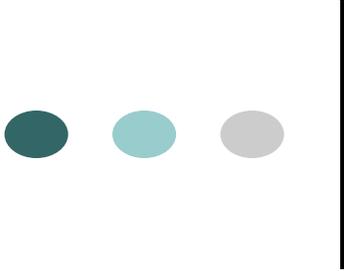
Suivre un plan prévisionnel →
refus du changement

Faible gestion de l'incertitude

(et du risque)

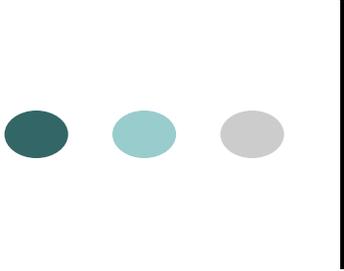
Peu de relation avec le Client

(effet tunnel)



Les Objectifs de l'agilité

- Trouver un compromis entre le **minimum de méthode** permettant de mener à bien les projets, tout en restant **adaptable et créatif**
- Accepter le **changement des besoins** et être capable d'y répondre de façon rapide et souple
- **Privilégier le code** plutôt que la documentation



Les moyens ?

- Utiliser un développement **itératif et incrémental**
- **Découper** le besoin et **prioriser**
- **Découper** la réalisation
 - Livrer fréquemment des *incréments* de produit
 - Accepter les *changements*
- Contrôler régulièrement l'avancement avec les **parties prenantes**

DEF ?

”

L'accent devrait être mis sur POURQUOI nous faisons quelque chose.

Edwards Deming

DEF Partie prenante

- C'est une personne qui est **intéressée** d'une façon ou d'une autre, **par le produit** réalisé par l'équipe :

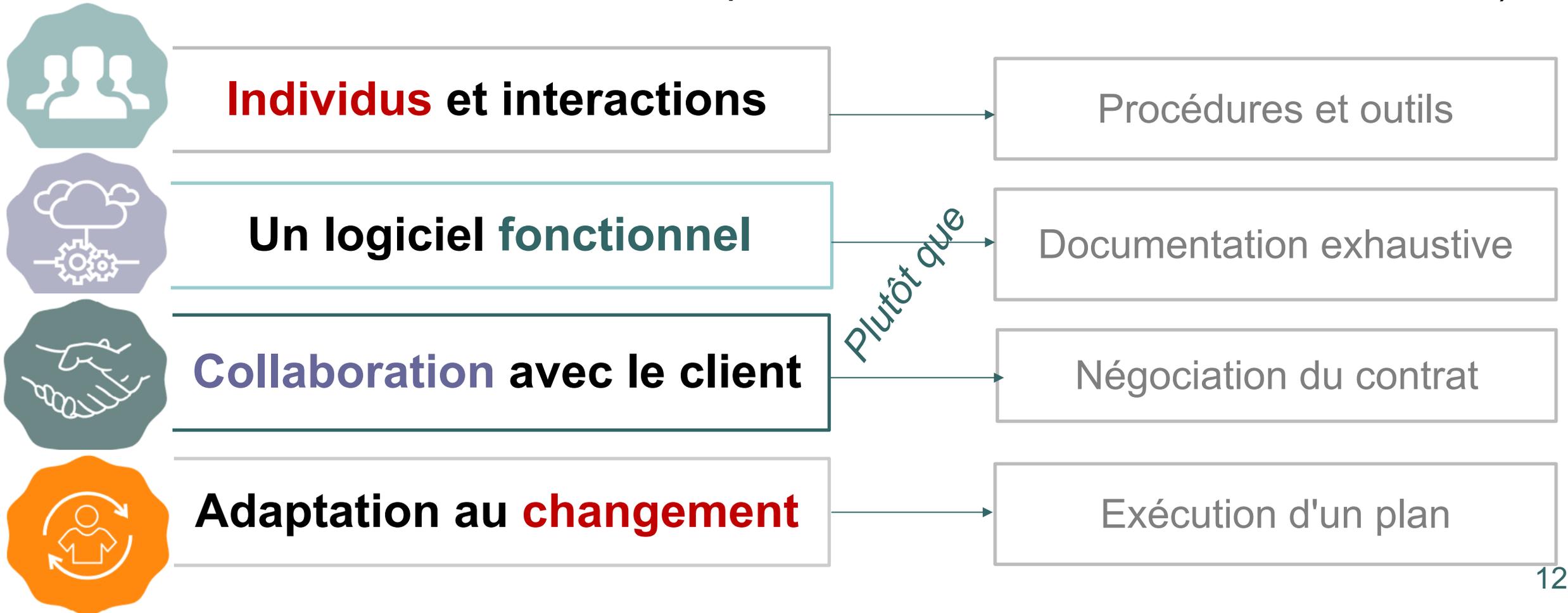


- Les utilisateurs, les clients, des représentants des utilisateurs
- Mais aussi des sponsors, des gens du marketing ou du commerce
- Aussi les managers, des personnes impliquées dans l'infrastructure, dans la qualité, le service des membres d'autres équipes, etc.

Ce sont les interlocuteurs privilégiés du Product Owner

Le « Manifeste Agile » 2001 : 4 valeurs de l'Agilité

(Privilégier les éléments de gauche bien que ceux de droite aient une certaine valeur)



Comparatif

Approche classique

Approche Agile

Périmètre (CDC)

Calendrier

Coût

Contraintes fixes

dirigé
par le
PLAN

dirigé
par la
VALEUR

Calendrier

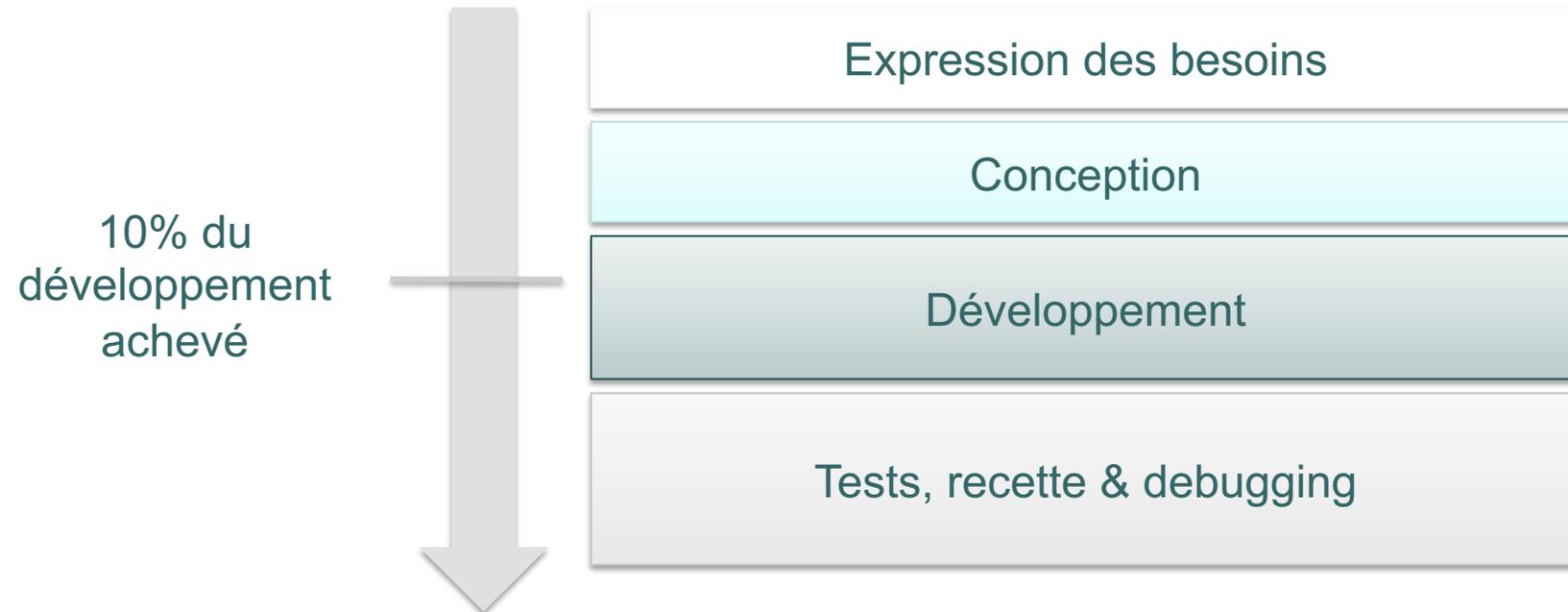
Coût

Périmètre (CDC)

Contraintes variables

L'agilité en 2 slides (1/2)

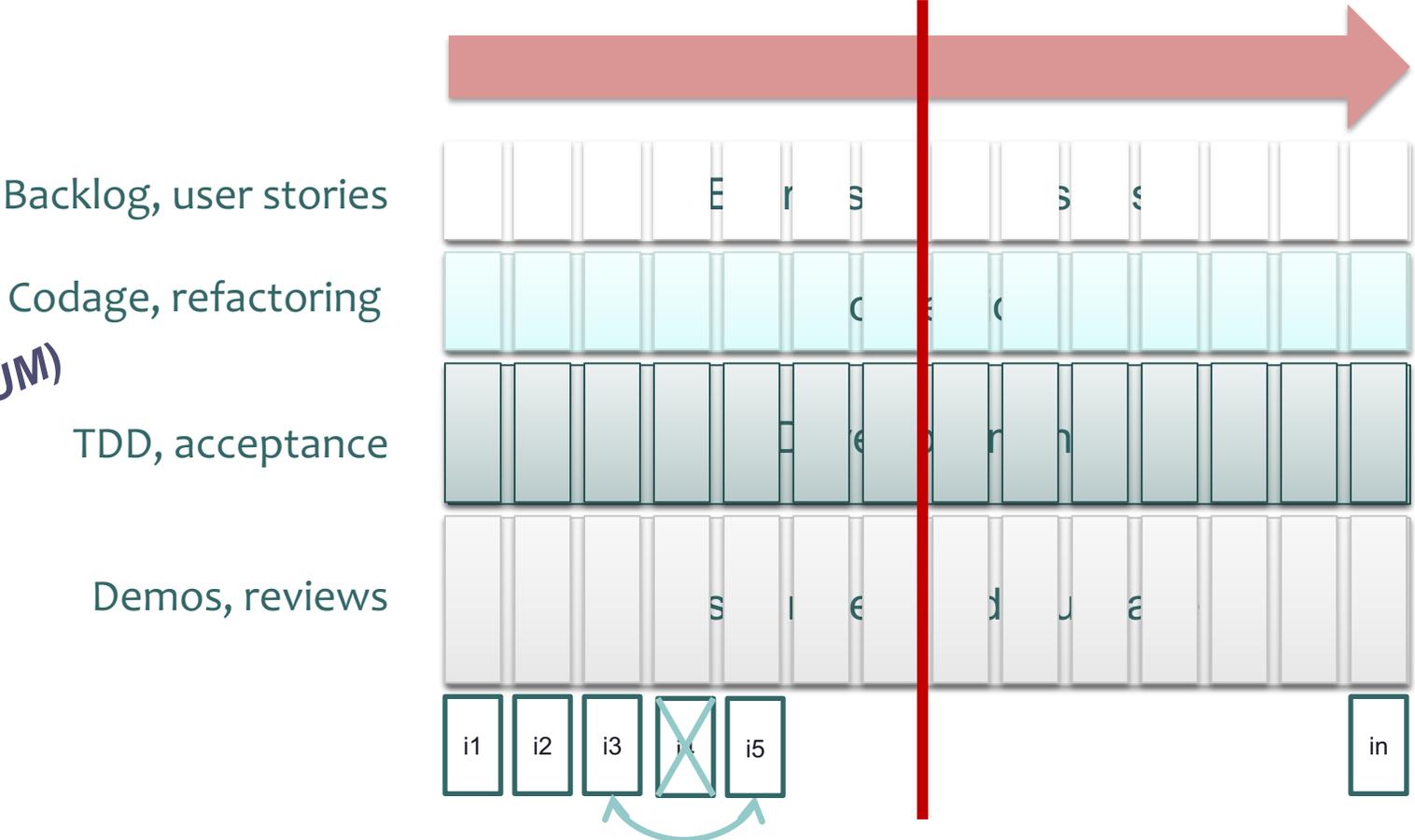
En développement classique, phases séquentielles



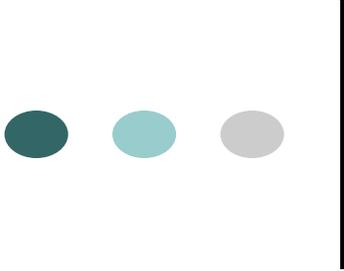
À mi parcours, le client ne voit statistiquement **que 10%** de son application

En développement agile : itérations

(phases de SCRUM)



À chaque itération, on a un incrément de produit **100%** fonctionnel
À mi parcours, on a **~80%** des fonctionnalités du logiciel



Facteurs de succès des méth. agiles

- Le client / l'utilisateur (ou son représentant) est impliqué **quotidiennement**
- Le management intermédiaire **soutient** l'équipe
- L'équipe est **auto-organisée**
- Les pratiques sont adaptées au mode incrémental
 - Des tests **automatisés** : rejoués souvent
 - Code **compréhensible** car va être sans doute modifié
 - Code **collectif**



Principaux freins à l'agilité



○ Résistance au changement

- Parfois on a de belles paroles, de beaux discours sur Scrum... Et en final les comportements, les structures et les processus restent **inchangés**
- Ex.: virage raté de la MACIF vers l'agilité en 2012
<https://www.linkedin.com/pulse/saffranchir-du-cycle-en-v-agile-canada-dry-ou-comment-maxime-blanc/>
 - Son éditeur, qu'elle attaquait en justice, a reçu 1,45M€ en final après jugement en 2016

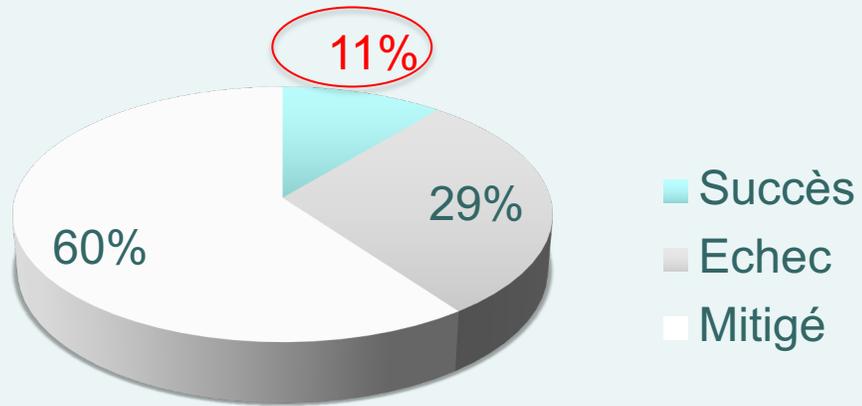
○ Un manque de **lâcher-prise** du management

- Accepter de *faire confiance* à l'équipe

Taux de réussite

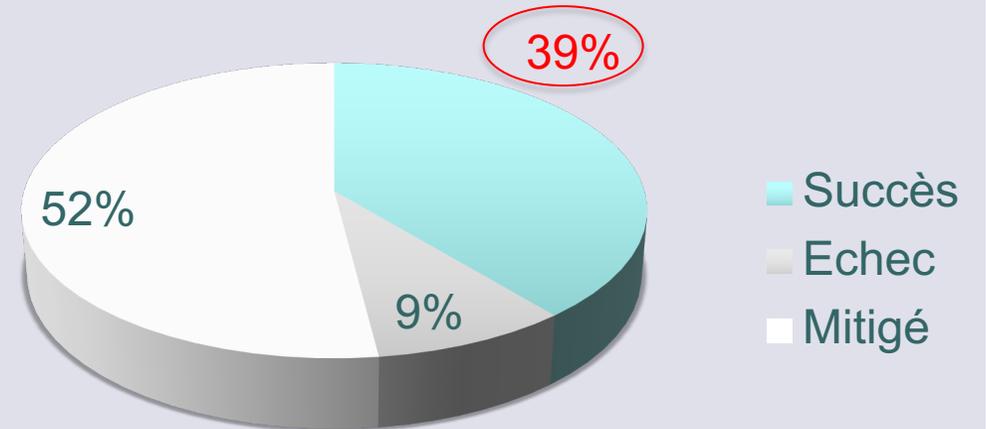
Source : the Standish Group,
« Chaos Report 2015 »

Approche Traditionnelle



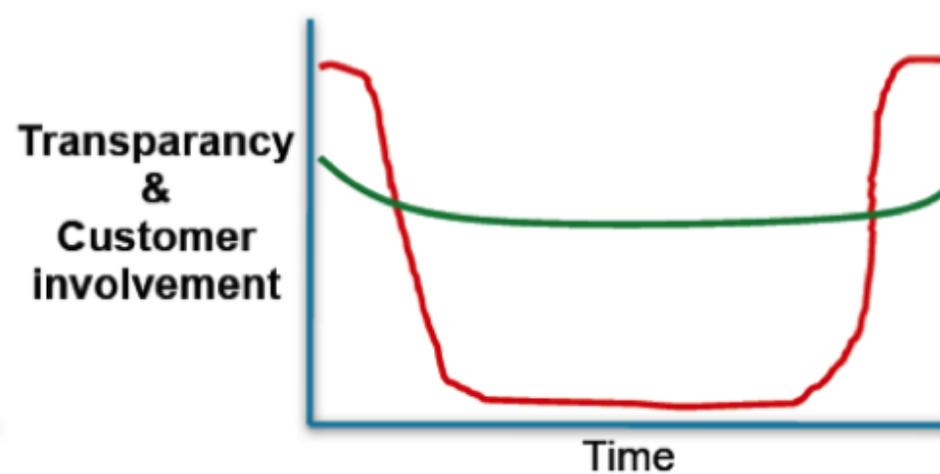
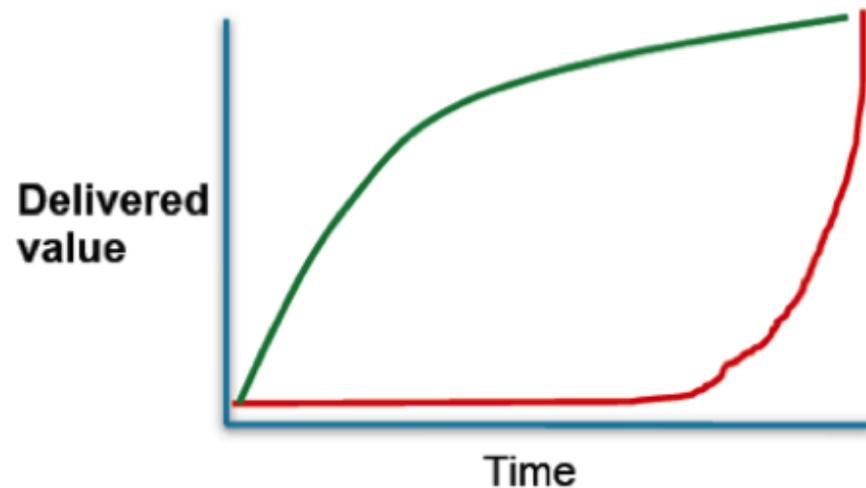
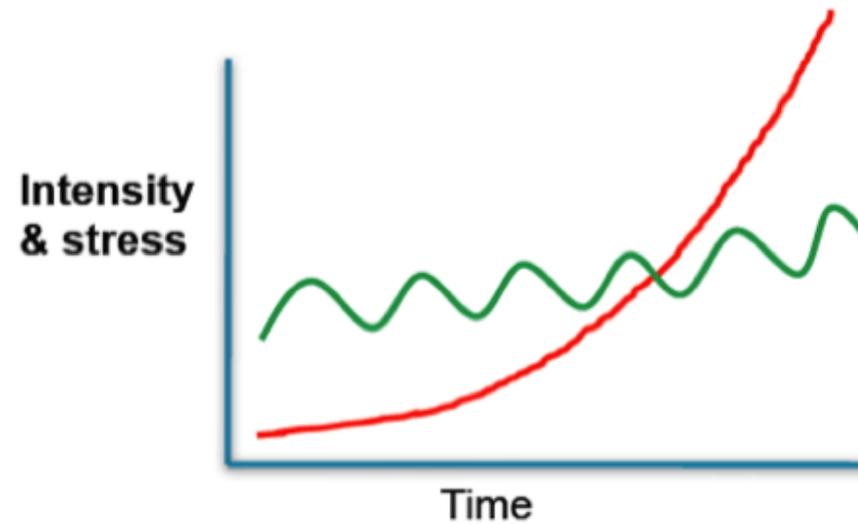
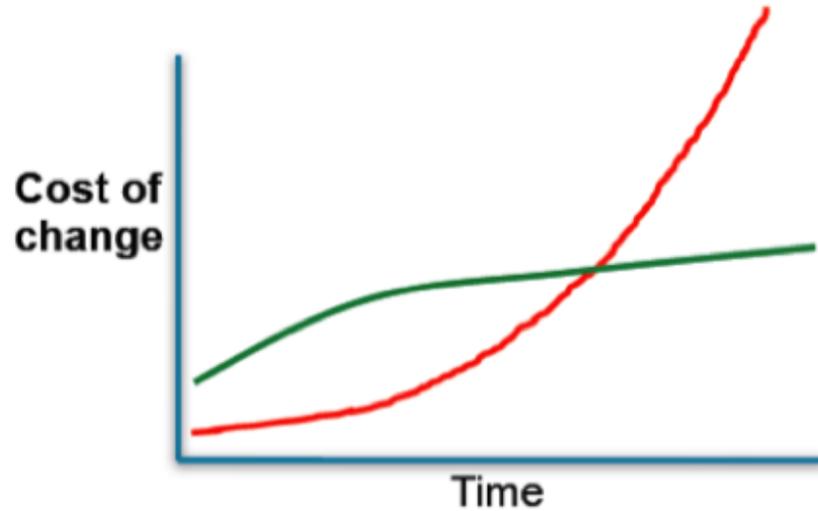
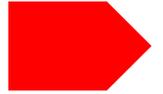
3x plus de projets **réussis**
3x moins d'**échecs**

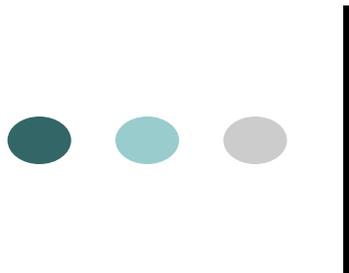
Approche Agile



Profils des projets

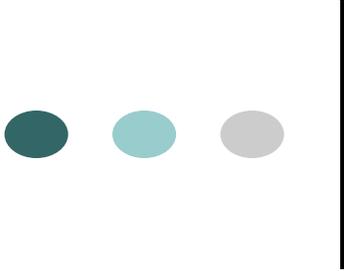
Agile vs. **Classique**





Si nécessaire...

ANNEXES



Attentes et bénéfices des Méth Agile

Attentes

1. **Accélérer la livraison** des logiciels et leur mise sur le marché (75%)
2. Bien gérer les **changements de priorité** dans les besoins des utilisateurs (64%)
3. Accroître la **productivité des équipes** (55%)
4. Mieux **aligner les attentes** du Métier et de la DSI (49%)
5. Améliorer la **qualité** du code produit (46%)

Bénéfices

1. Bien gérer les **changements de priorité** dans les besoins des utilisateurs (71%)
2. Avoir une bonne visibilité des projets
3. Mieux **aligner les attentes** du Métier et de la DSI (65%)
4. **Accélérer la livraison** des logiciels et leur mise sur le marché (62%)
5. Accroître la **productivité des équipes** (61%)



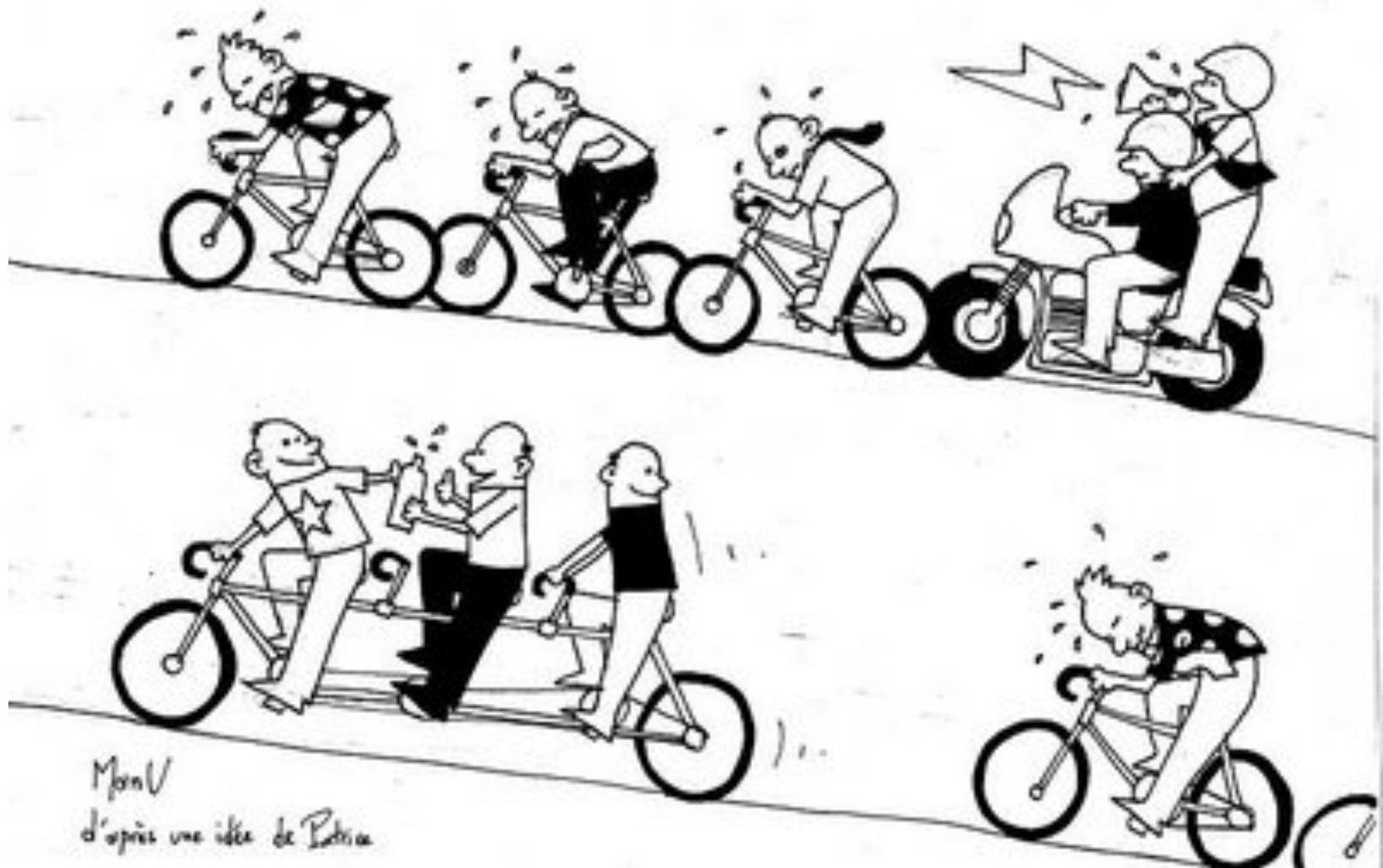
Heureux que le trou ne soit pas de notre côté

Considérer l'organisation dans sa globalité

Accepter le collectif



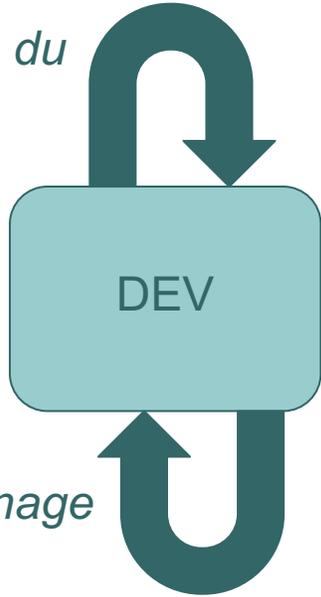
La puissance de l'équipe



Cycle complet du Dév Logiciel



Définition du besoin



(étape vue à l'IUT)

(le reste dans la vraie vie)

Serveur de développement

Serveur de qualification

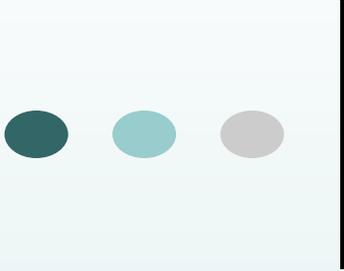
Serveur de production

Le cycle peut ainsi être long avant qu'on découvre que les utilisateurs finaux sont mécontents... L'agilité vise à rapprocher au mieux le VRAI besoin. Cycles courts, itérations, adaptation

- ● ●

Test unitaire vs. Test d'intégration





Je retiens :

- Constat **d'échec** des approches classiques
- Les **objectifs** des Méthodes Agiles
- Les **moyens** proposés
- Les **4 valeurs** des Méthodes Agiles
- Le **taux réussite** par rapport aux approches classiques
- Comprendre les **schémas** des profils Agile vs. Classique