

# Agilité, les fondamentaux

Emmanuel CHAUVET

1

# Les fondamentaux de l'agilité

# Objectifs des méthodes agiles

Réduire le cycle de vie du logiciel et accélérer son développement

Développer une version minimale *via* l'intégration de fonctionnalités


Processus itératif et adaptatif basé sur une écoute client

Tests tout au long du cycle de développement

# Origines des méthodes agiles



Instabilité de l'environnement technologique



Client incapable de définir les besoins de manière exhaustive en début de projet



Adaptation aux changements durant le projet (évo/modif des specs en cours de route)



2001 : Manifeste agile constitué par 17 personnes (rupture avec GdP classique)

# Les origines en image



**Le besoin exprimé par l'utilisateur**



**Ce que le chef de projet a compris**



**Comment l'analyste l'a schématisé**



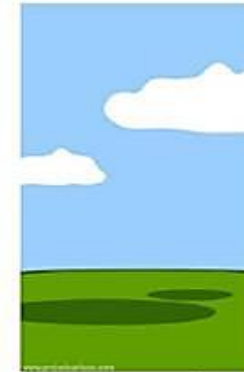
**Comment le programmeur l'a écrit**



**Ce que l'équipe de test a reçu**



**Comment le Business Consultant l'a décrit**



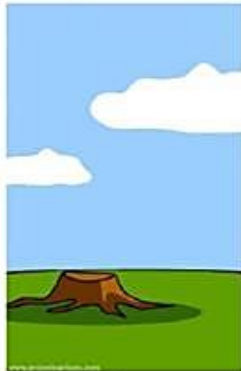
**La documentation livrée**



**L'application livrée**



**Ce qui a été facturé au client**



**Ce que gère le support technique**



**La communication faite auprès des utilisateurs**



**La date réelle de la livraison**



**Ce dont l'utilisateur avait vraiment besoin**



**L'avis des utilisateurs**



**Les mesures correctives apportées**

# Le Manifeste Agile

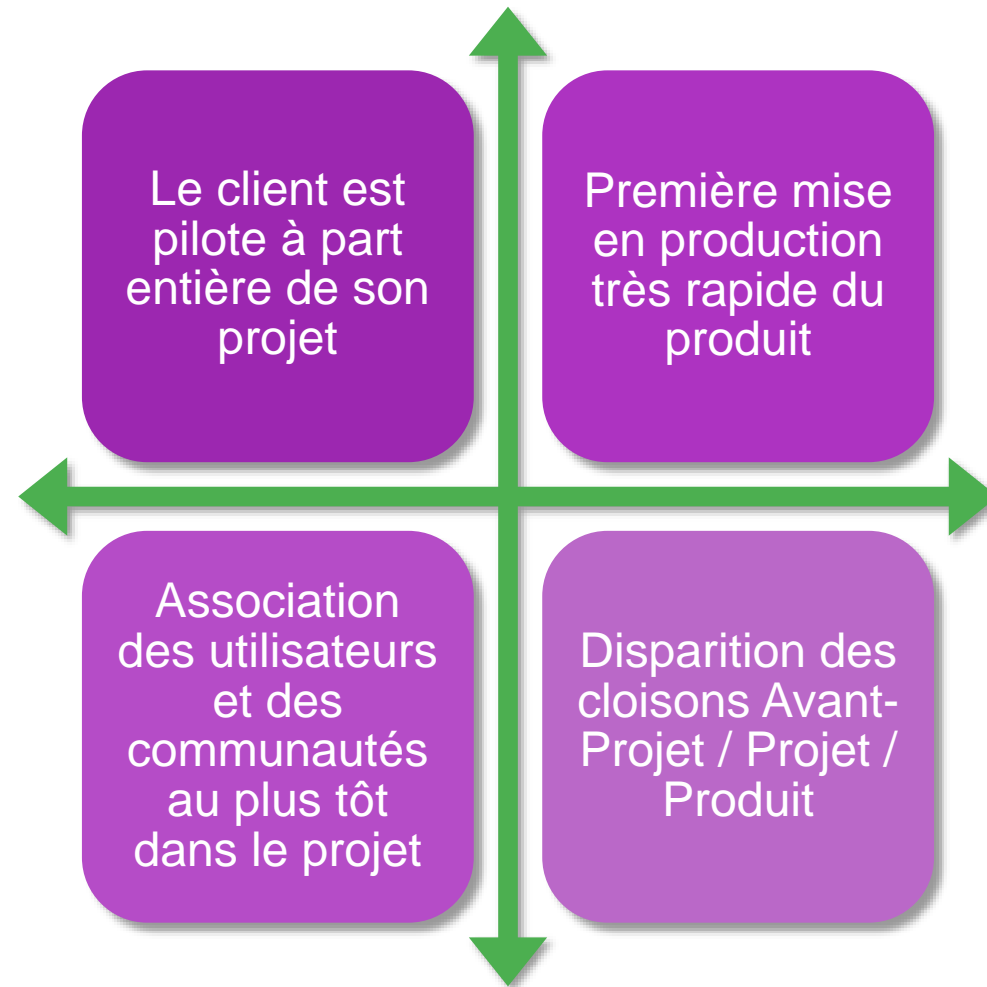
Nous découvrons comment mieux développer des logiciels  
par la pratique et en aidant les autres à le faire.  
Ces expériences nous ont amenés à valoriser :

**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

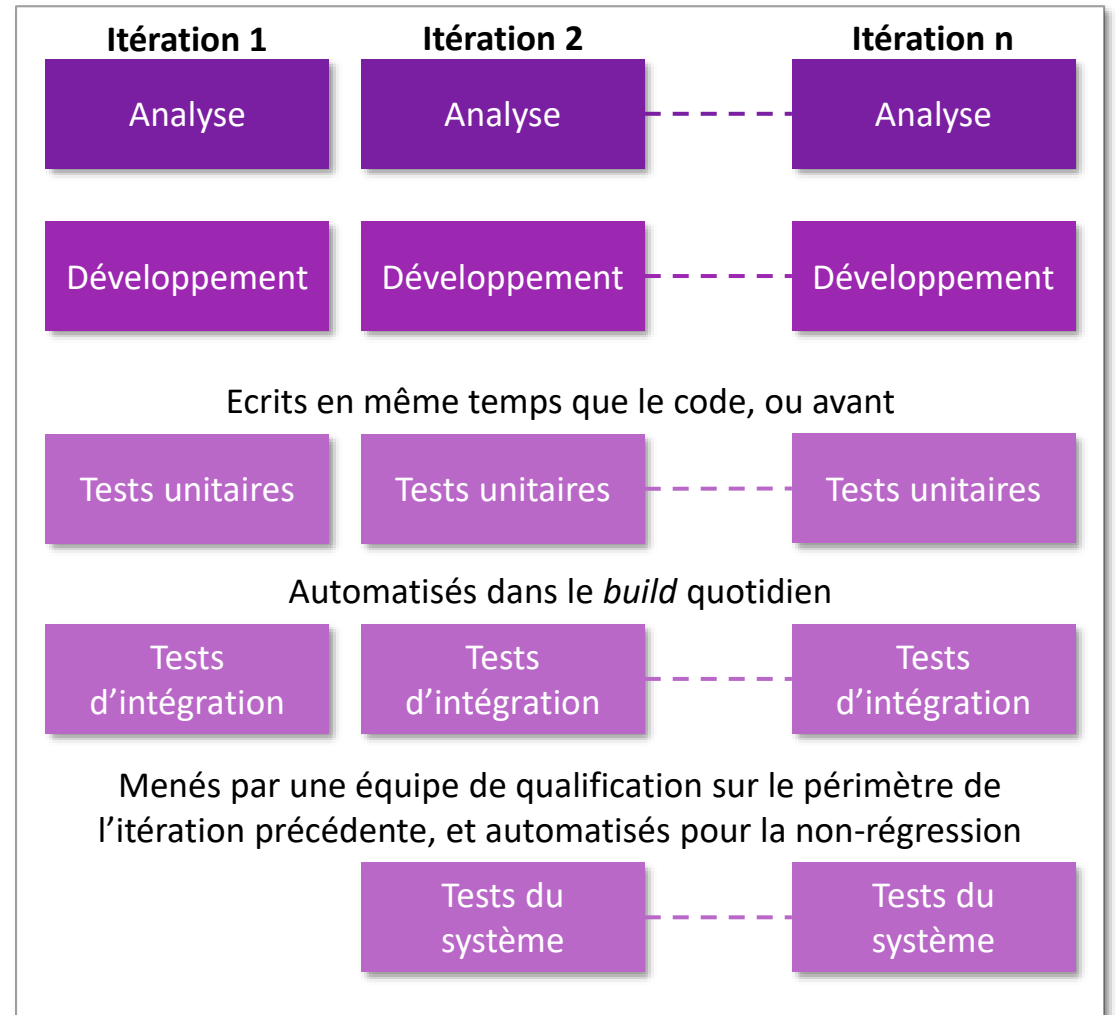
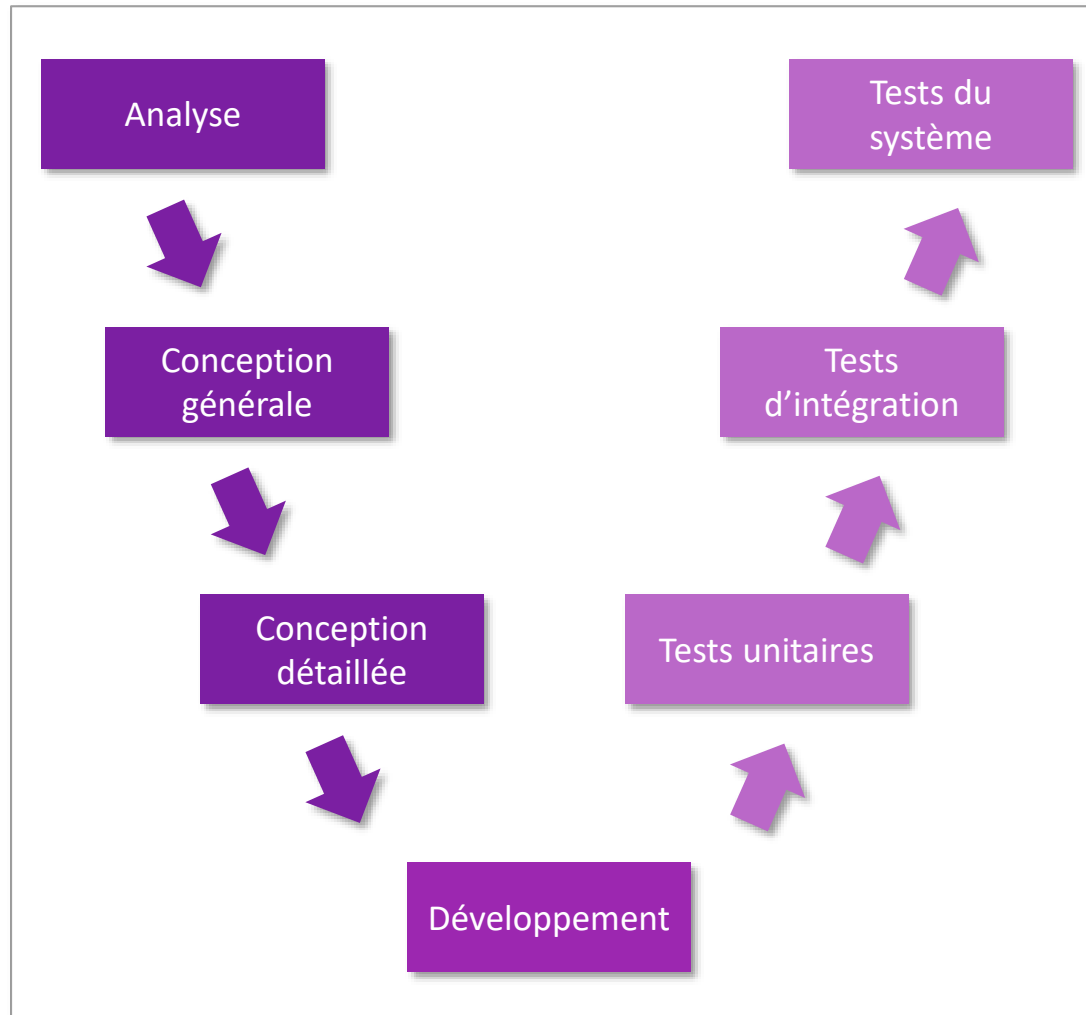
Nous reconnaissons la valeur des seconds éléments,  
mais privilégions les premiers.

<http://agilemanifesto.org/iso/fr/manifesto.html>

# Bénéfices côté client



# Cycle en V vs Méthodes agiles





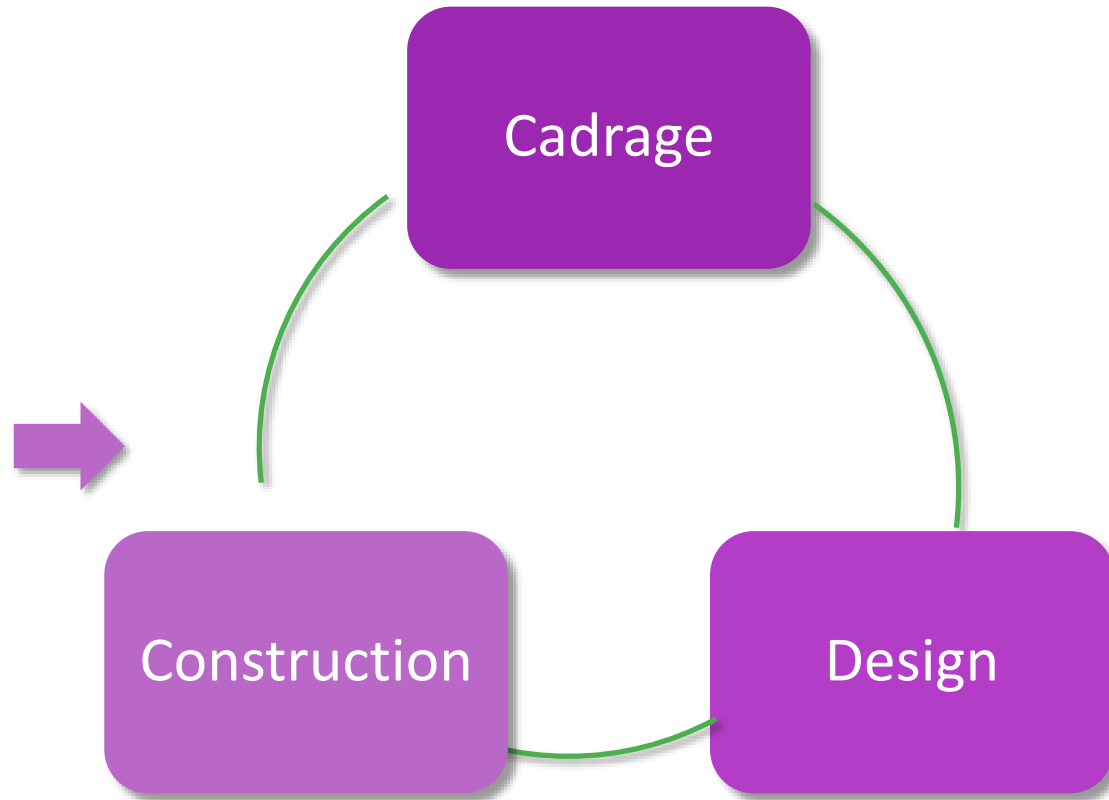
2

## Inventaire (partiel) des méthodes agiles

# RAD – *Rapid Application Development*

James Martin (1991)

Cycle de développement  
court (90 à 120 jours max)



# DSDM – *Dynamic Software Development Method*

## Basé sur RAD (UK, 1994)

Implication

des utilisateurs durant tout le cycle de développement

Autonomie

L'équipe projet a un pouvoir de décision sur l'évolution des besoins

Visibilité du résultat

Application livrée le plus souvent possible pour avoir un feed-back rapide

Adéquation

Livrer une application en adéquation avec le besoin du client

Développement itératif et incrémental

Evolution du développement basée sur le feed-back des utilisateurs

Réversibilité

Toute modification effectuée durant le développement est réversible

Synthèse

Schéma directeur défini au préalable pour fixer le périmètre du projet

Tests

continus durant le développement pour garantir le fonctionnement de l'application

Coopération

Acteurs du projet faisant preuve de souplesse sur les modifications demandées

# UP – *Unified Process*



Développement **itératif** et **incrémental**



Segmentation en **phases courtes**



**Livraison** systématique en **fin de phase** d'une nouvelle version **incrémentée**



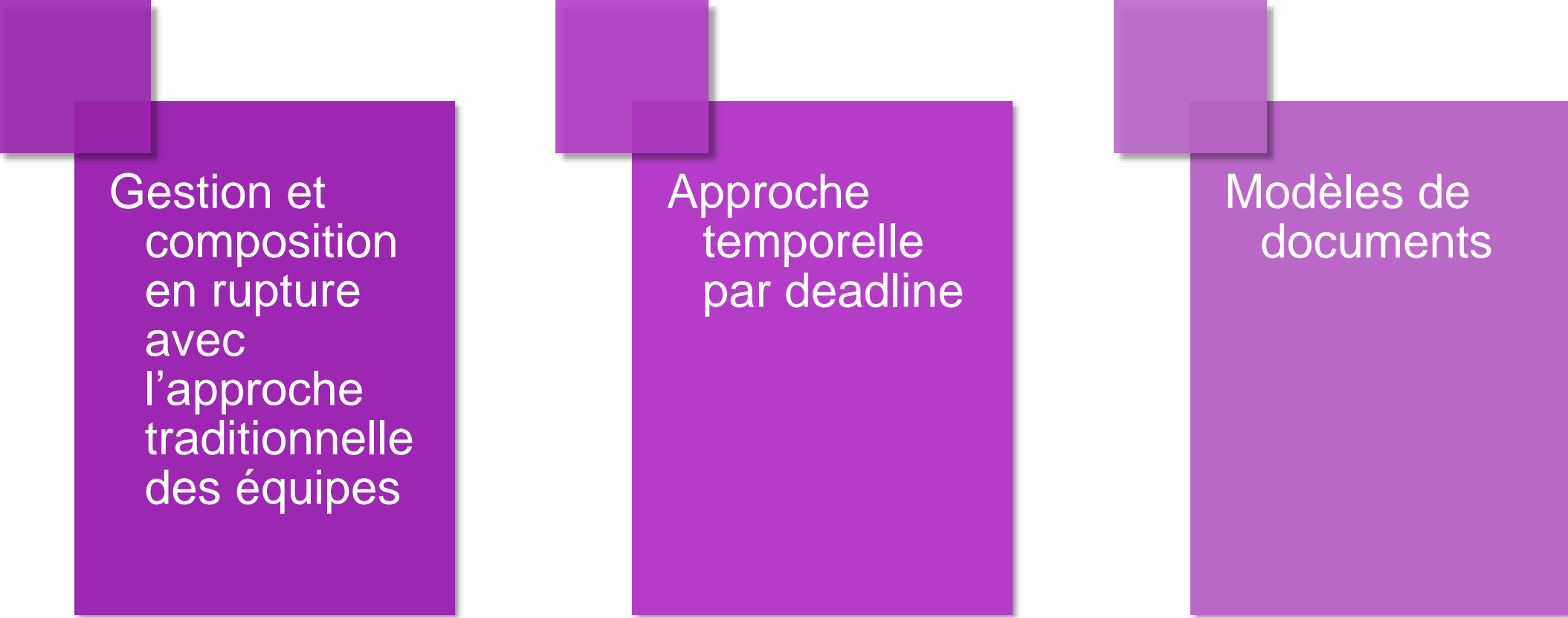
Modélisation **UML** pour l'**architecture** (fonctionnelle, logicielle et physique)



**Cas d'utilisation** (*use case*) pour les **besoins** et **exigences** des utilisateurs

# RUP – *Rational Unified Process*

Implémentation d'UP proposée par Rational Software (racheté depuis par IBM)



Gestion et  
composition  
en rupture  
avec  
l'approche  
traditionnelle  
des équipes

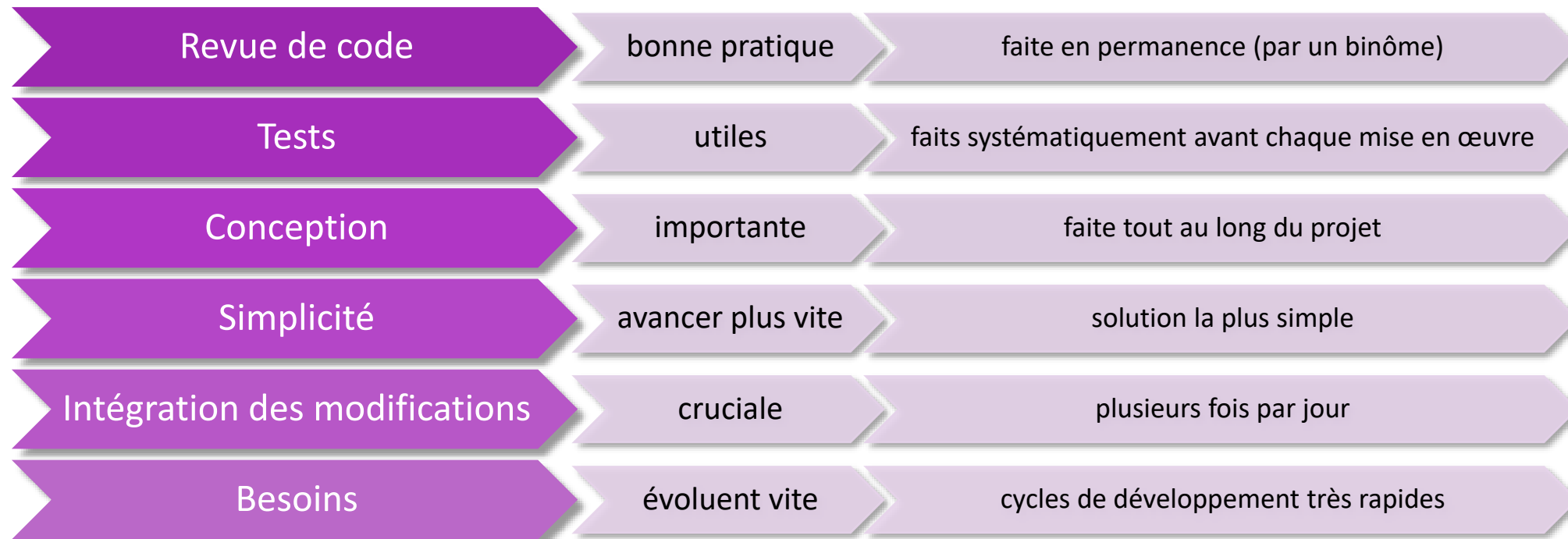
Approche  
temporelle  
par deadline

Modèles de  
documents

# XP – *eXtreme Programming*

Kent Beck, Ward Cunningham, Ron Jeffries & Palleja Xavier (1999)

Systematisation extrême de pratiques pré-existantes



# Synthèse et recommandations

Ne pas choisir une  
méthode

mais s'inspirer des existantes

Garder en tête les  
principes agiles

- Le client est au centre du projet
- Moins de documentation mais plus de livraisons
- Deadline et découpage en phases courtes
- Gestion des équipes
- Tests tout au long du projet

Une question à se poser

« Le client est-il agile ? »