

Introduction aux méthodes agiles

Clé d'inscription au cours

Agile_2025

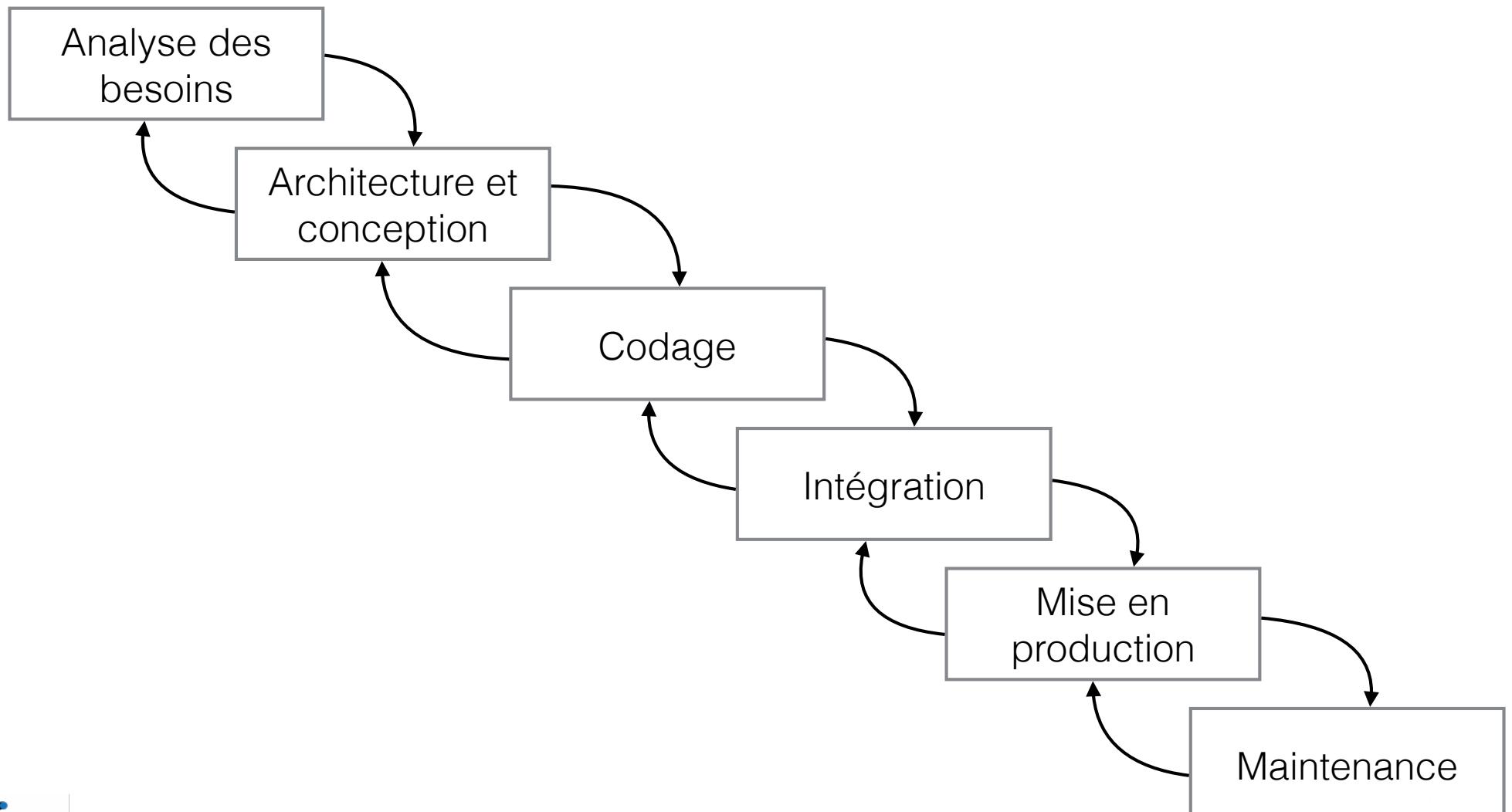
Préambule

Méthodes de conception du logiciel

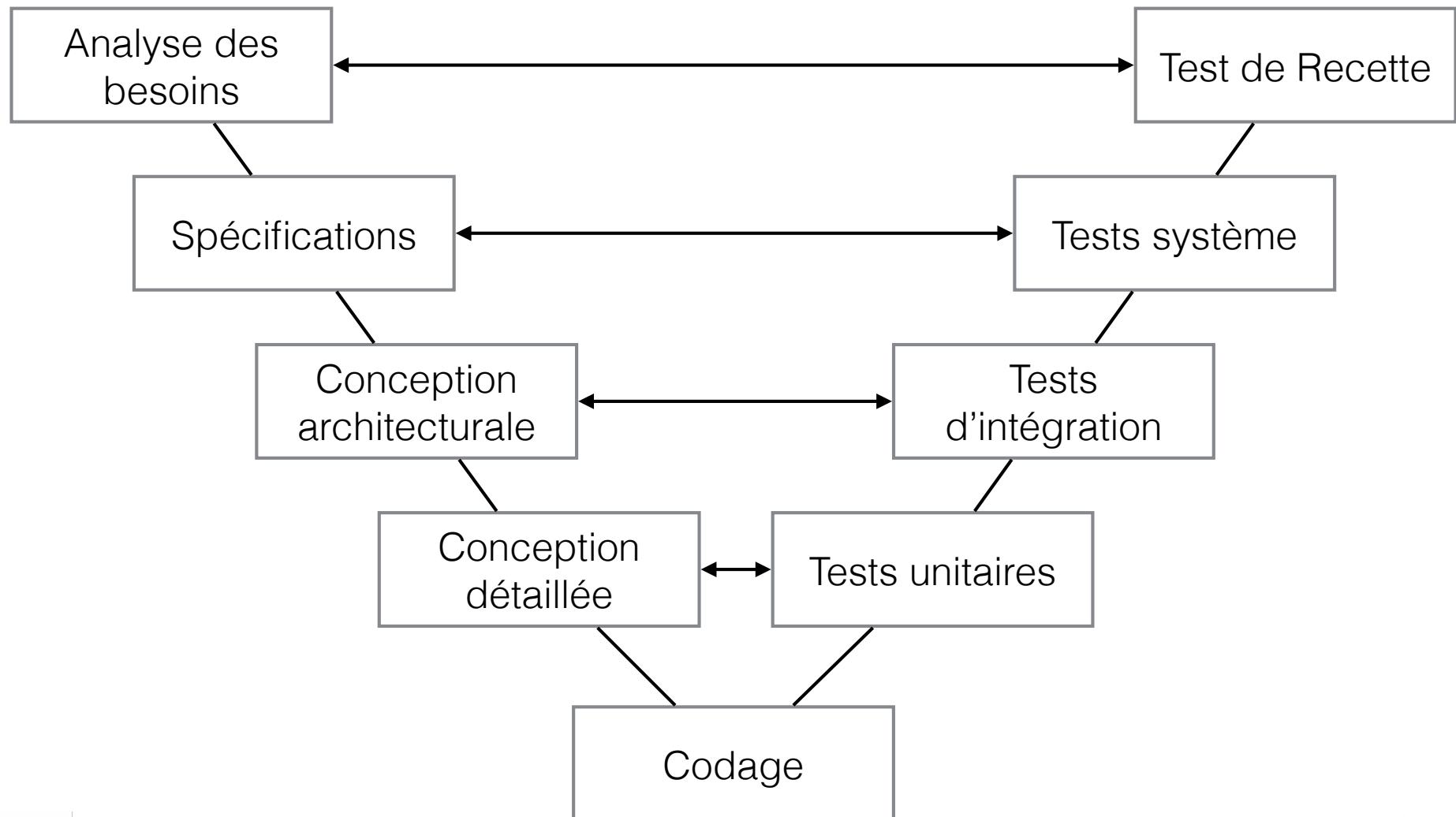
Vie d'un produit

- Identification du besoin
- Création du produit
- Livraison
- Maintenance

Cascade (1966)



Cycle en V (1980)



Critique

- Validation d'étapes sur revues de documents
- Production de nombreux documents
- Etapes séquentielles
- Difficulté de prise en compte du changement

BTP vs Informatique

- Durée de vie du produit
- Evolution du besoin
- Périmètre fonctionnel constant / changeant

D'autres cycles...

- Cycle par prototypage (années 80)
- Cycle en Y (années 90)

Problèmes

- Attentes et retards
- Transmission d'informations
- Processus et fonctionnalités inutiles
- Effet tunnel

How Projects Really Work (version 1.5)

Create your own cartoon at www.projectcartoon.com



How the customer explained it



How the project leader understood it



How the analyst designed it



How the programmer wrote it



What the beta testers received



How the business consultant described it



How the project was documented



What operations installed



How the customer was billed



How it was supported



iSwing



What the customer really needed

Méthodes Agiles (2000)

Manifeste agile

- <http://www.agilemanifesto.org>
- <http://www.agilemanifesto.org/iso/fr/>
- **4 valeurs**

Manifeste pour le développement Agile de logiciels

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 privilégiions les premiers.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Principes sous-jacents

- <http://www.agilemanifesto.org/iso/fr/principles.html>
- **12 principles**

Principes sous-jacents au manifeste

Nous suivons ces principes:

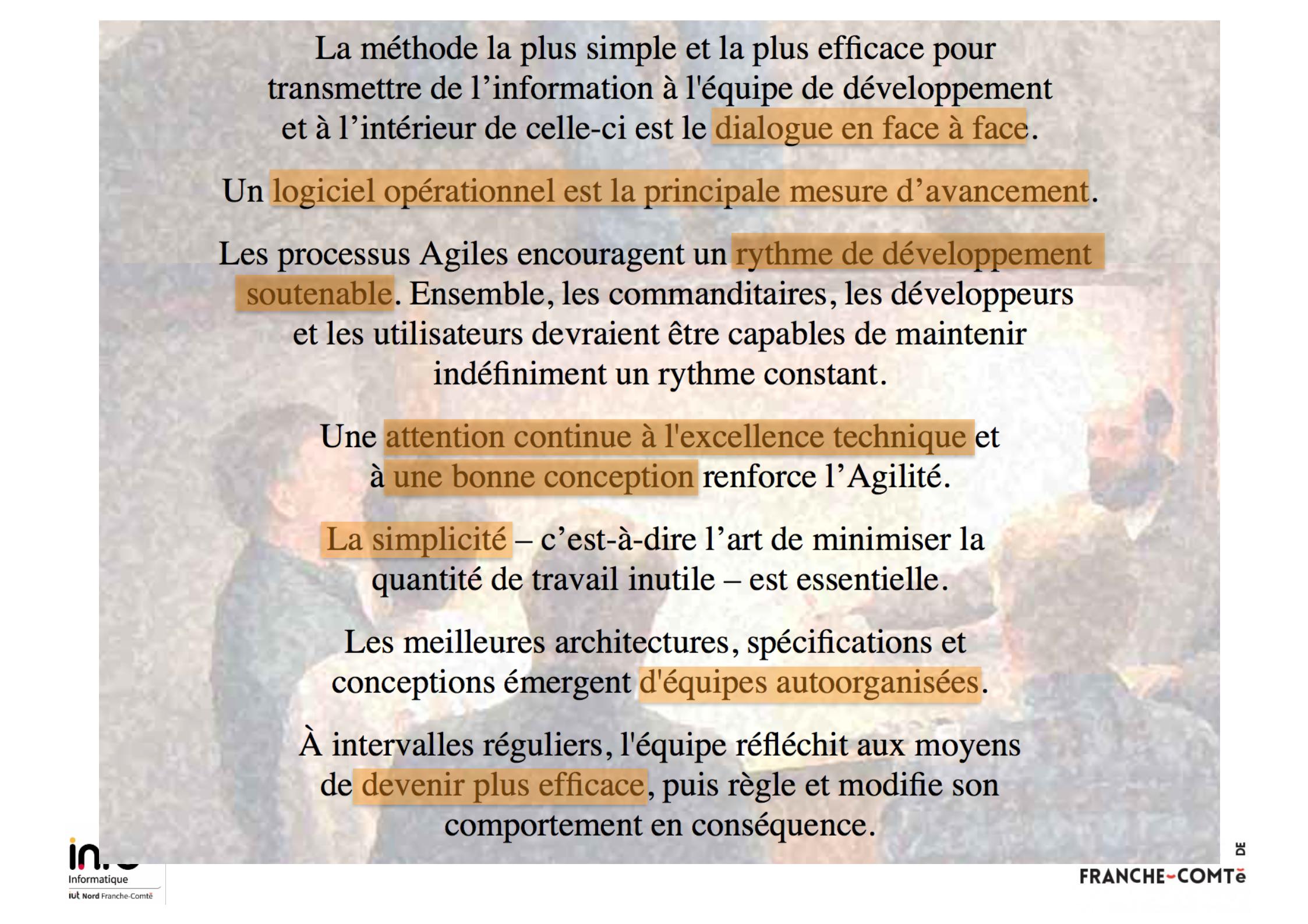
Notre plus haute priorité est de **satisfaire le client** en livrant rapidement et régulièrement des fonctionnalités à grande valeur ajoutée.

Accueillez positivement les changements de besoins, même tard dans le projet. Les processus Agiles exploitent le changement pour donner un avantage compétitif au client.

Livrez fréquemment un logiciel opérationnel avec des cycles de quelques semaines à quelques mois et une préférence pour les plus courts.

Les utilisateurs ou leurs représentants et les développeurs doivent **travailler ensemble quotidiennement** tout au long du projet.

Réalisez les projets avec des personnes motivées. Fournissez-leur l'environnement et le soutien dont ils ont besoin et **faites-leur confiance** pour atteindre les objectifs fixés.



La méthode la plus simple et la plus efficace pour transmettre de l'information à l'équipe de développement et à l'intérieur de celle-ci est le dialogue en face à face.

Un logiciel opérationnel est la principale mesure d'avancement.

Les processus Agiles encouragent un rythme de développement soutenable. Ensemble, les commanditaires, les développeurs et les utilisateurs devraient être capables de maintenir indéfiniment un rythme constant.

Une attention continue à l'excellence technique et à une bonne conception renforce l'Agilité.

La simplicité – c'est-à-dire l'art de minimiser la quantité de travail inutile – est essentielle.

Les meilleures architectures, spécifications et conceptions émergent d'équipes autoorganisées.

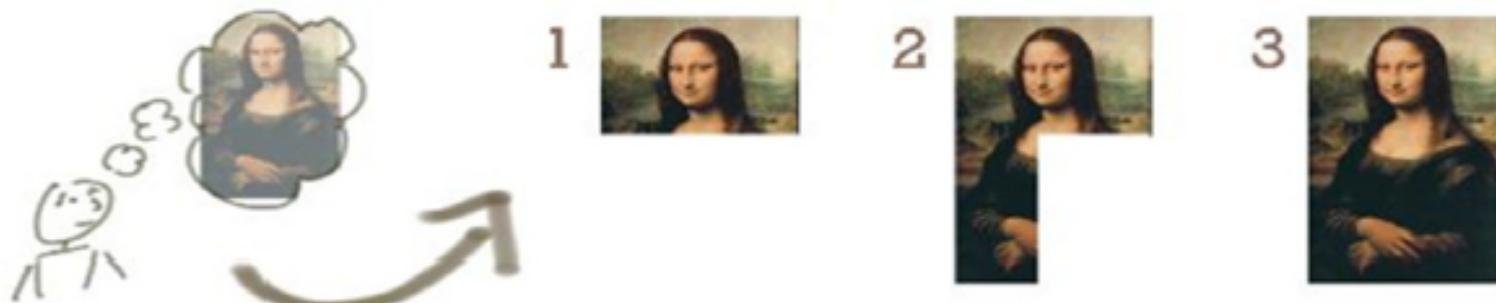
À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis règle et modifie son comportement en conséquence.

Méthodes

- XP : eXtrem Programming
(Kent Beck, Ward Cunningham, Ron Jeffries)
- SCRUM (Ken Schwaber, Jeff Sutherland)

Cycle agile

Incrémental



Itératif

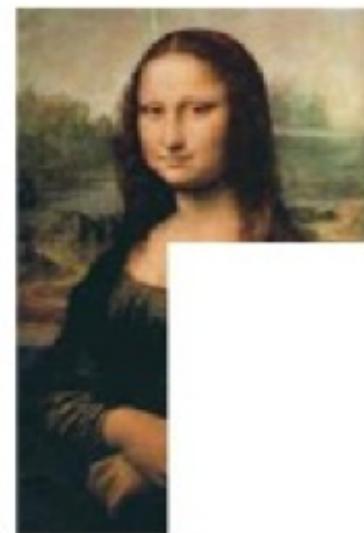


Niveau Itérations

1



2



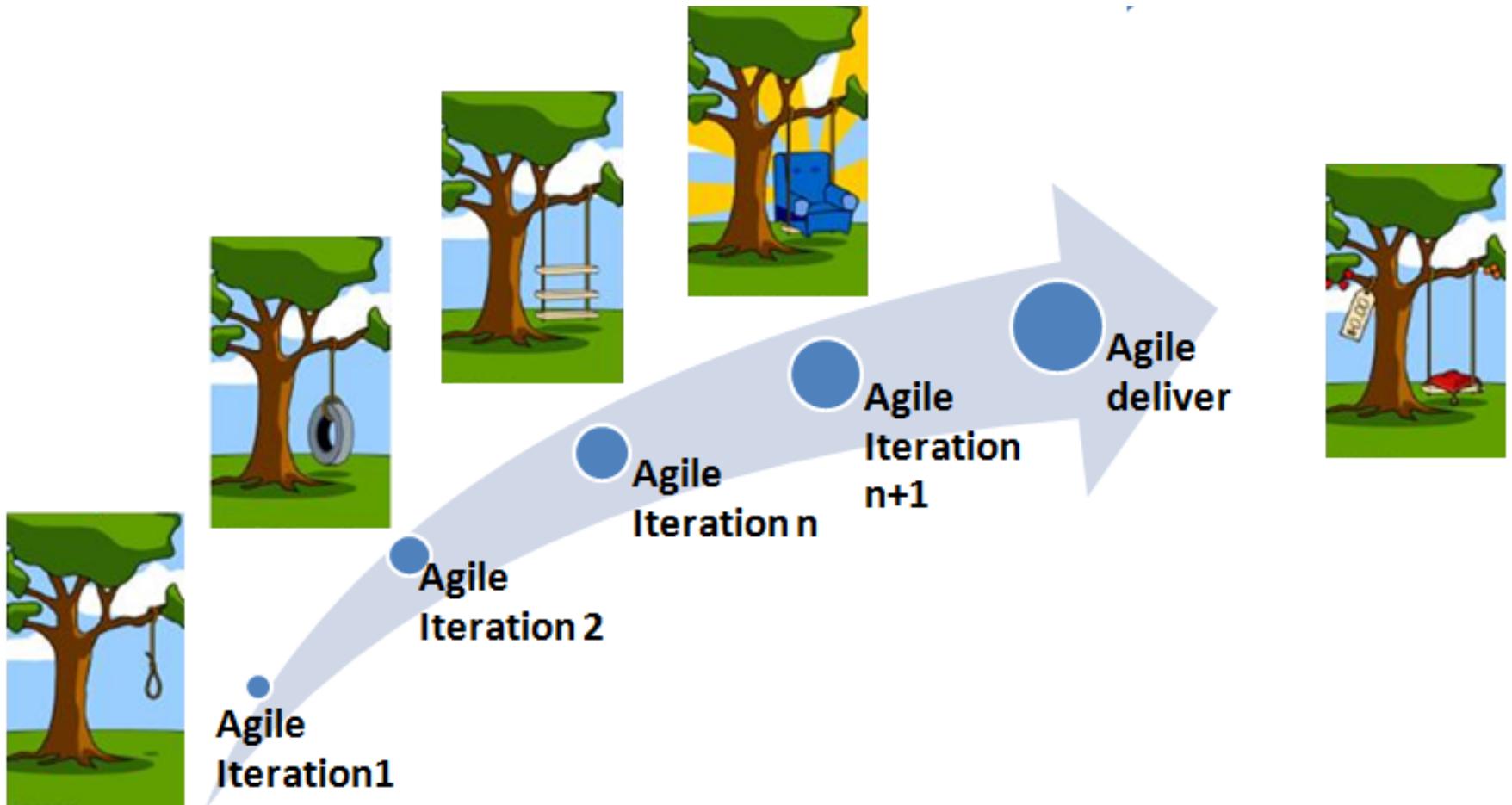
3



Contenu itération 1

Contenu itération 2

Contenu itération 3

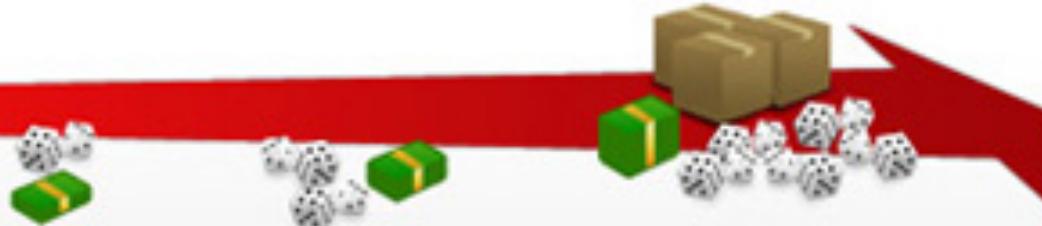


DEVELOPMENT: AGILE VS WATERFALL

where the risk lies

WATERFALL

time



Deliverable Product



Chance of failure



Project Run Rate

Agile development uses an iterative approach to reduce risk and create products that are "launchable" throughout development.

DEVELOPMENT: AGILE VS WATERFALL

where the risk lies

WATERFALL

time



AGILE

time



Deliverable Product



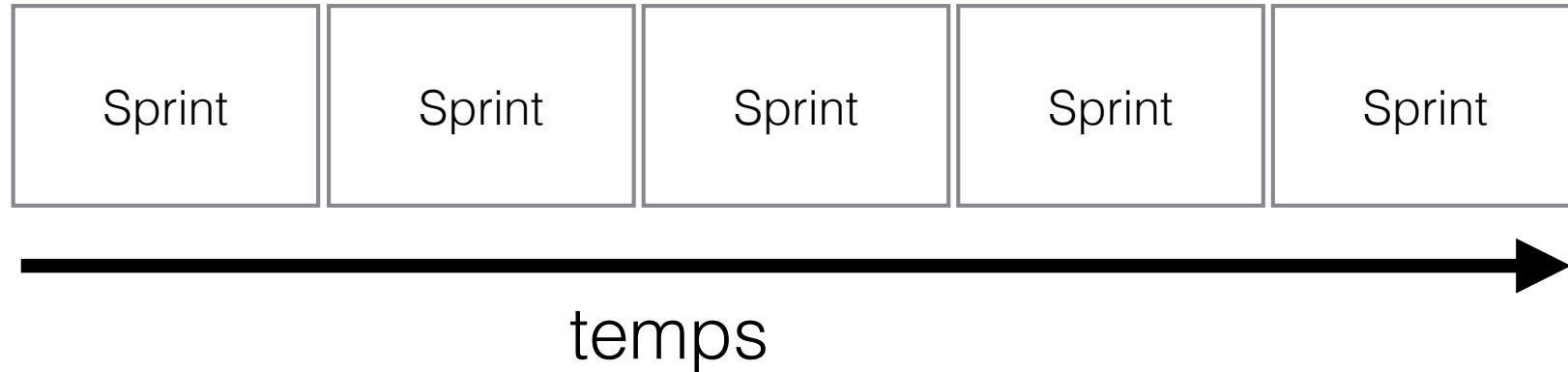
Chance of failure



Project Run Rate

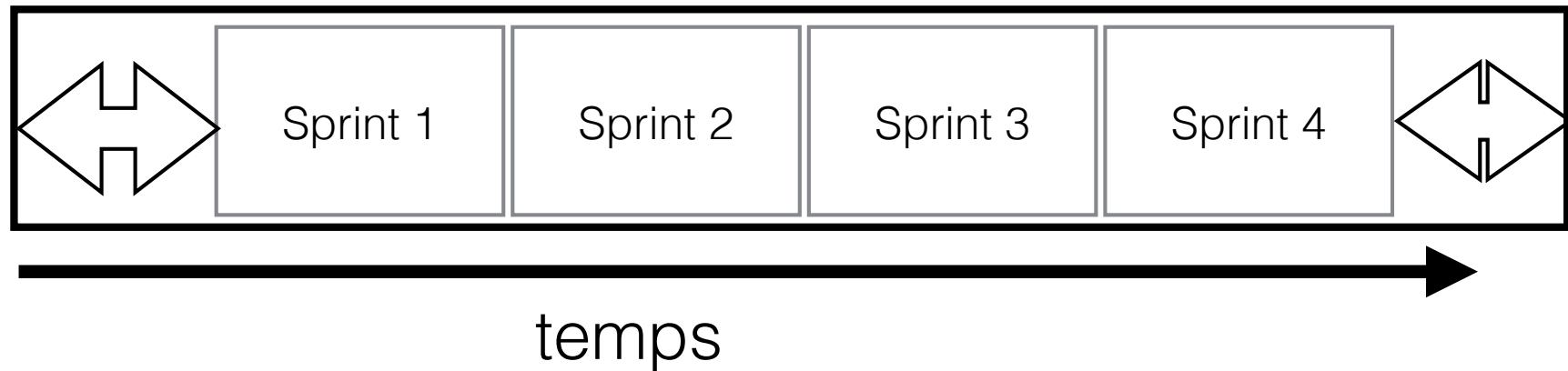
Agile development uses an iterative approach to reduce risk and create products that are "launchable" throughout development.

Le Sprint



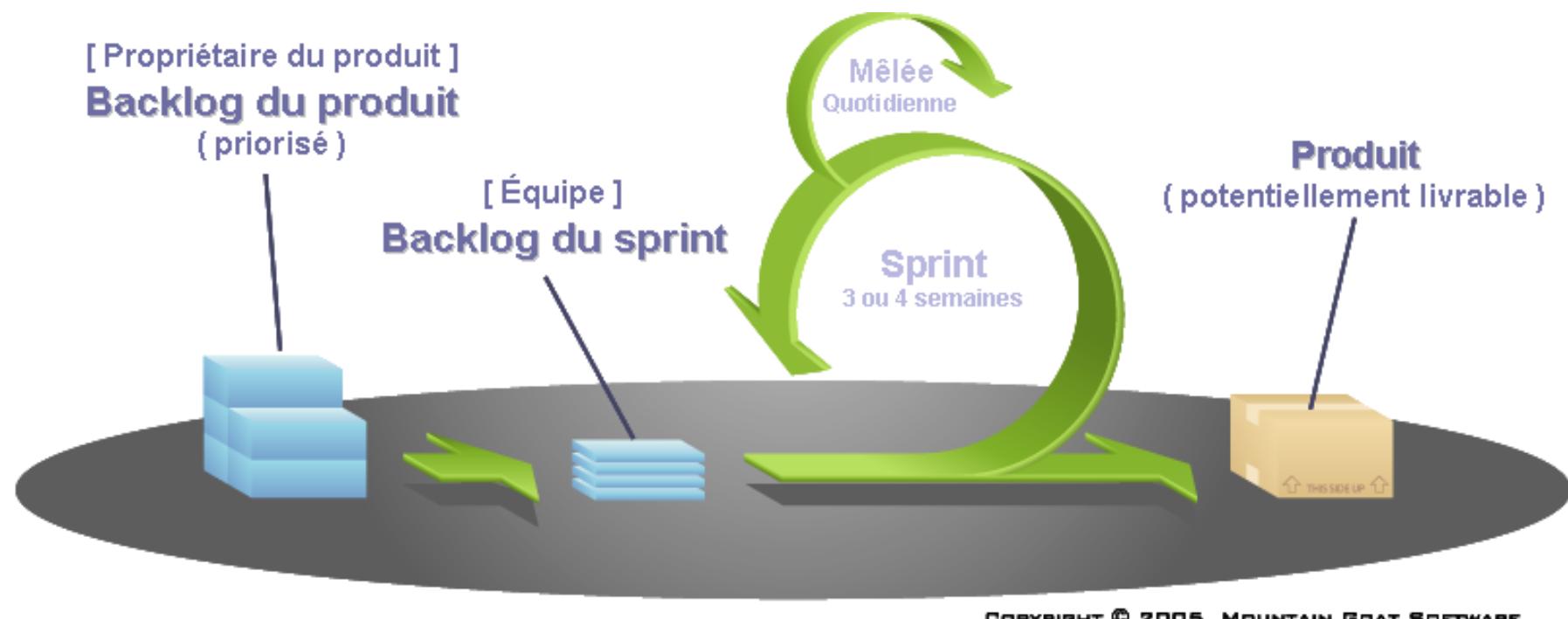
- Timebox
- Specification, Architecture, Codage, Test

La Release



- Début de release
 - mise en place équipe, environnement de dev
 - préparation du backlog initial
- Fin de release (pas toujours)
 - Mise en prod à chaud, packaging, mise à dispo en téléchargement...

Structure d'un Sprint



SCRUM

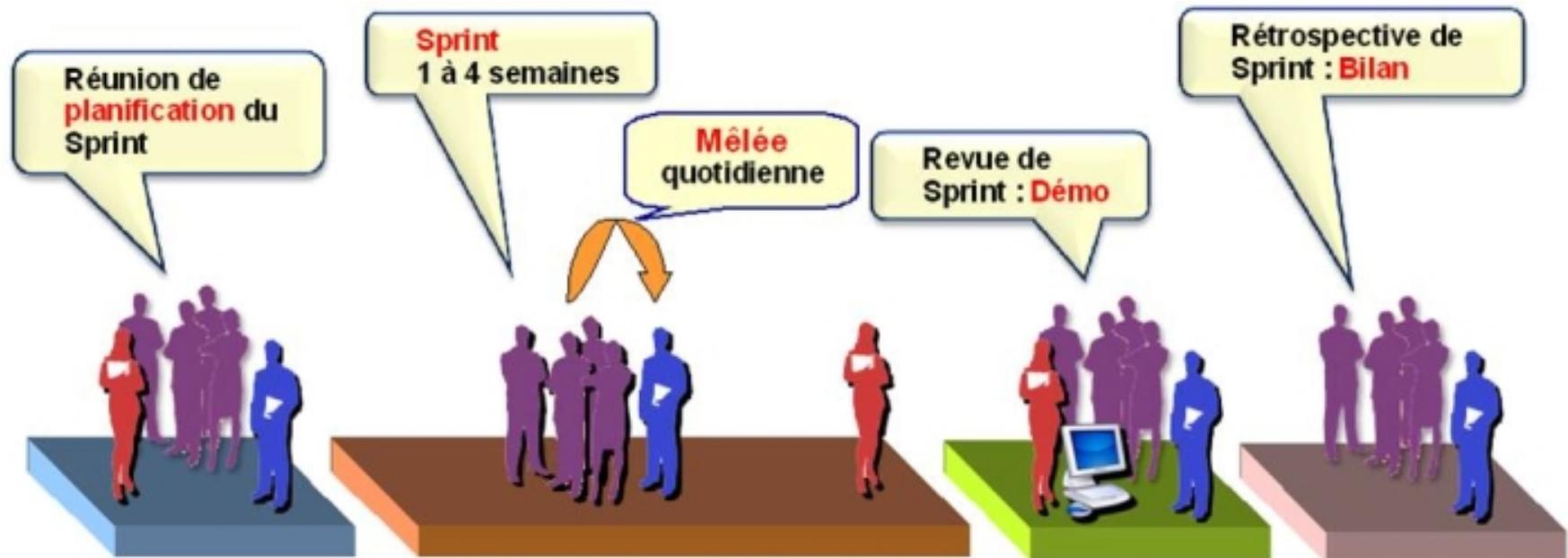
Rôles

- Product Owner
- ScrumMaster
- Développeur
- Parties prenantes

Artefacts

- Backlog
- Plan de release
- Plan de sprint
- Burndown
- Définition de fini
- Liste des obstacles

Cérémonial



- Product owner
- Scrummaster
- Équipe Scrum

Product Owner

- Faire partager la vision du produit
- Définir de contenu du produit
 - backlog
- Planifier la vie du produit

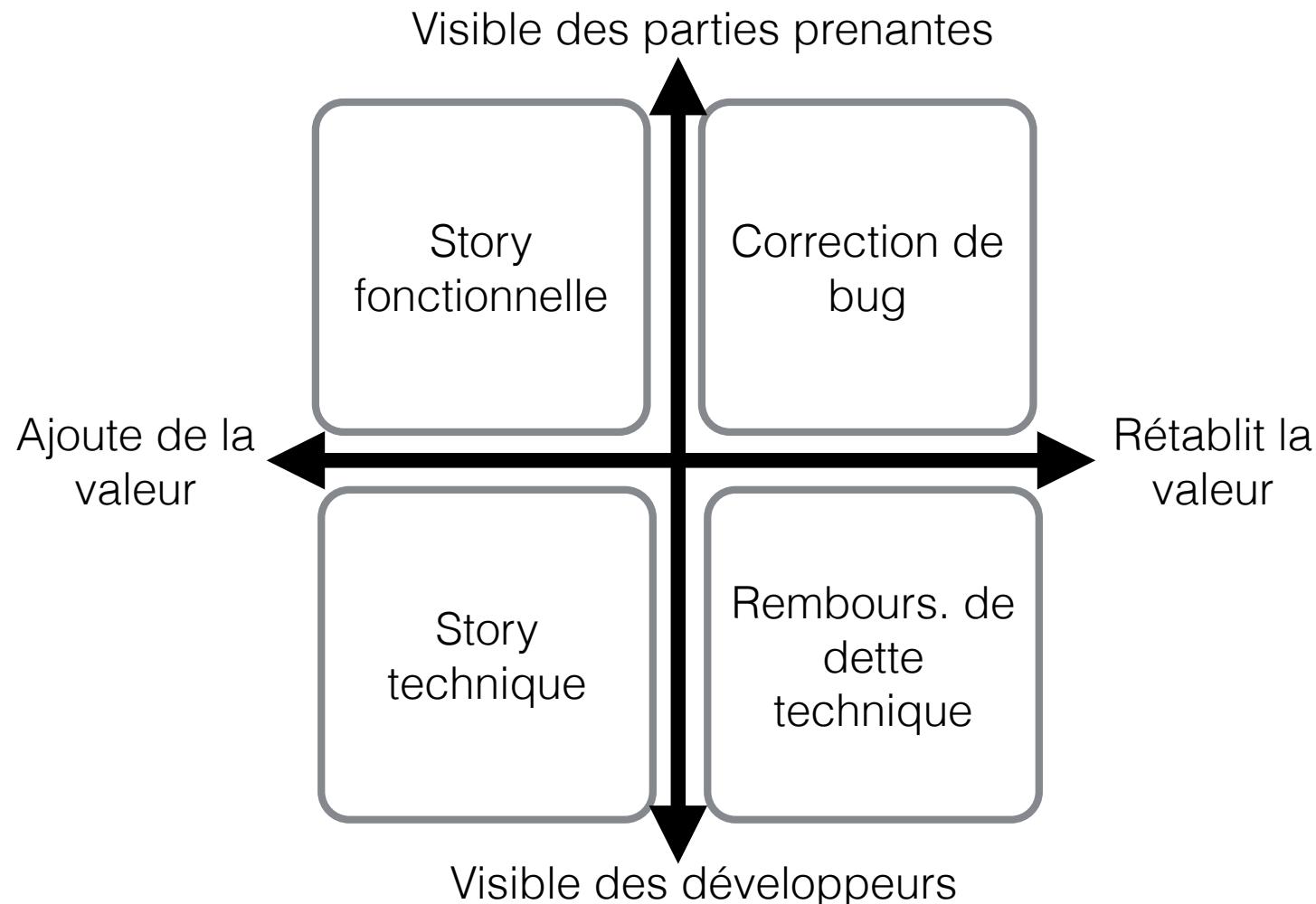
ScrumMaster

Responsabilité essentielle : aider l'équipe à réaliser les travaux demandés par le Product Owner, en appliquant Scrum

Backlog & Stories

- Backlog : contient l'ensemble des activités de l'équipe de dev
- Story : un élément du backlog

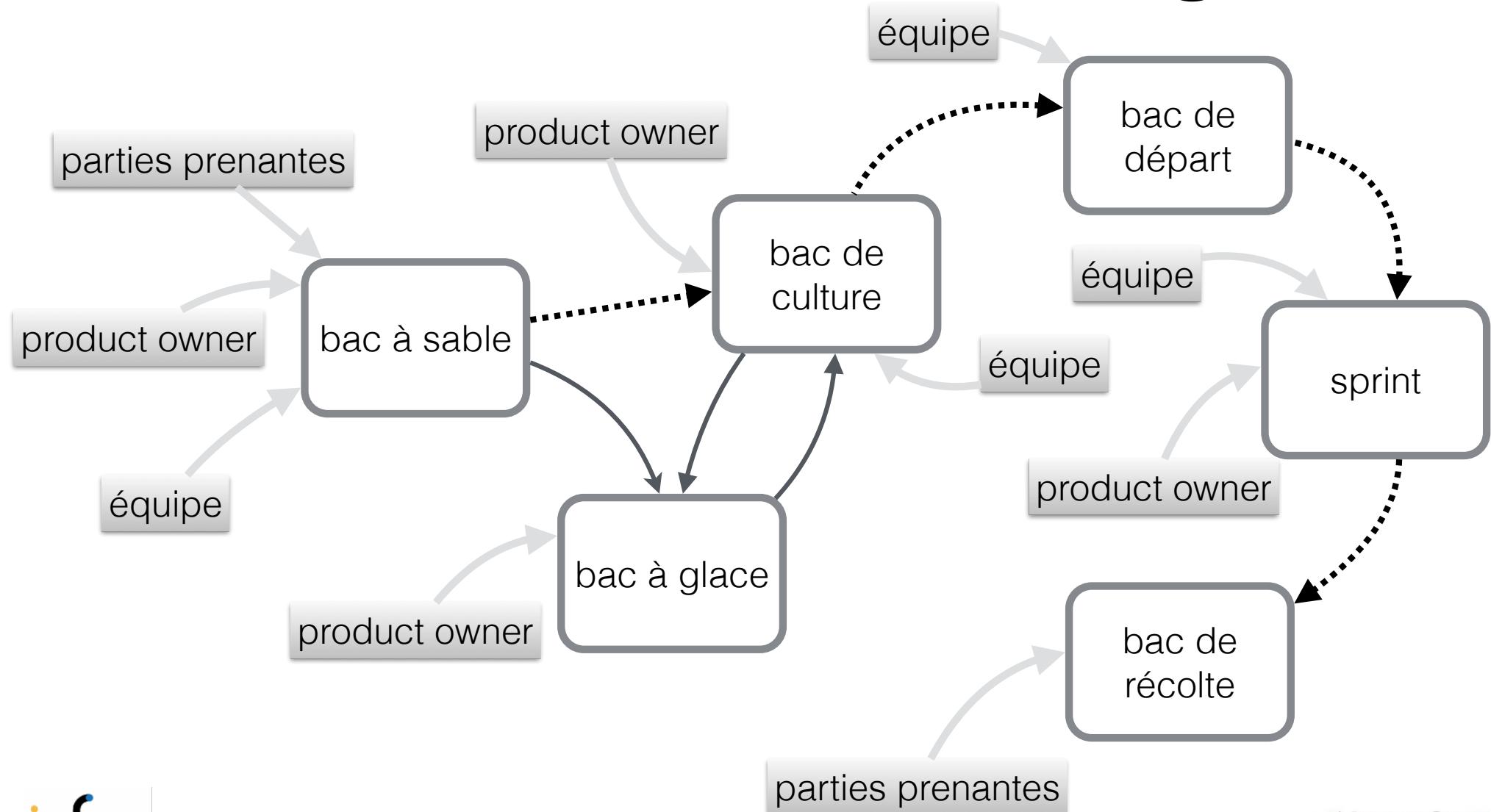
4 types de story



Les états d'une story



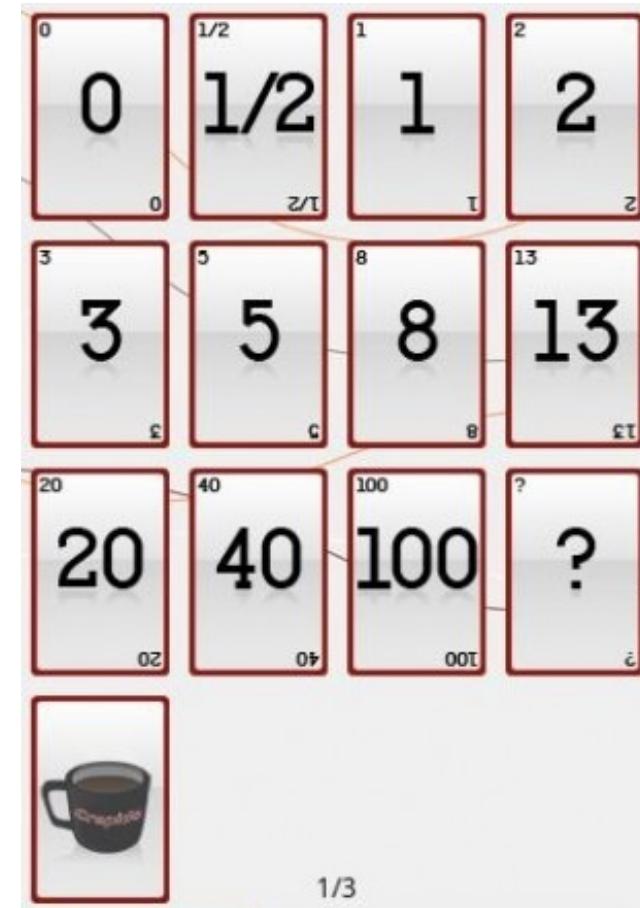
Bacs du backlog



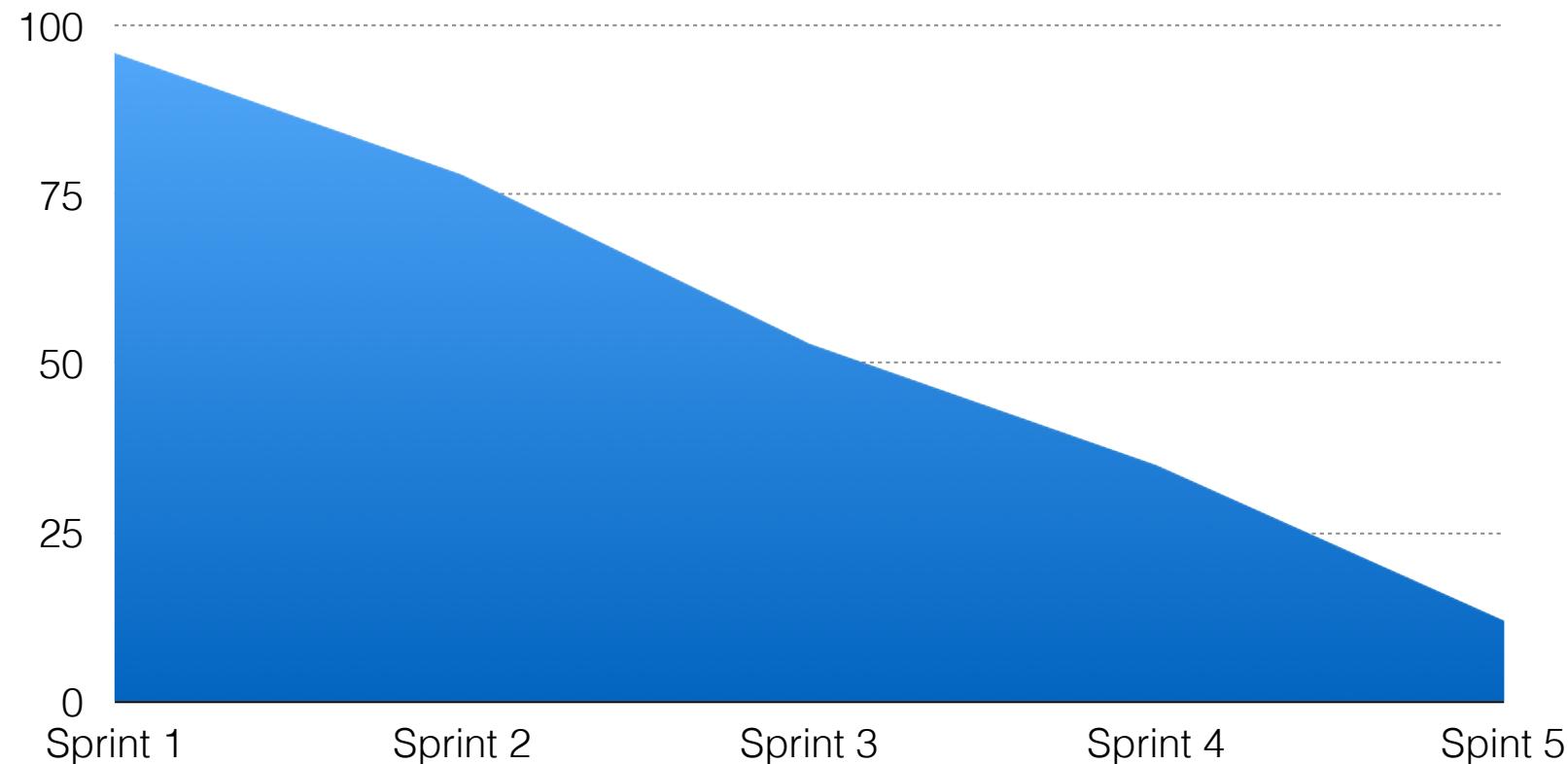


Planifier

- backlog
- planning poker



Burndown chart



Burnup chart

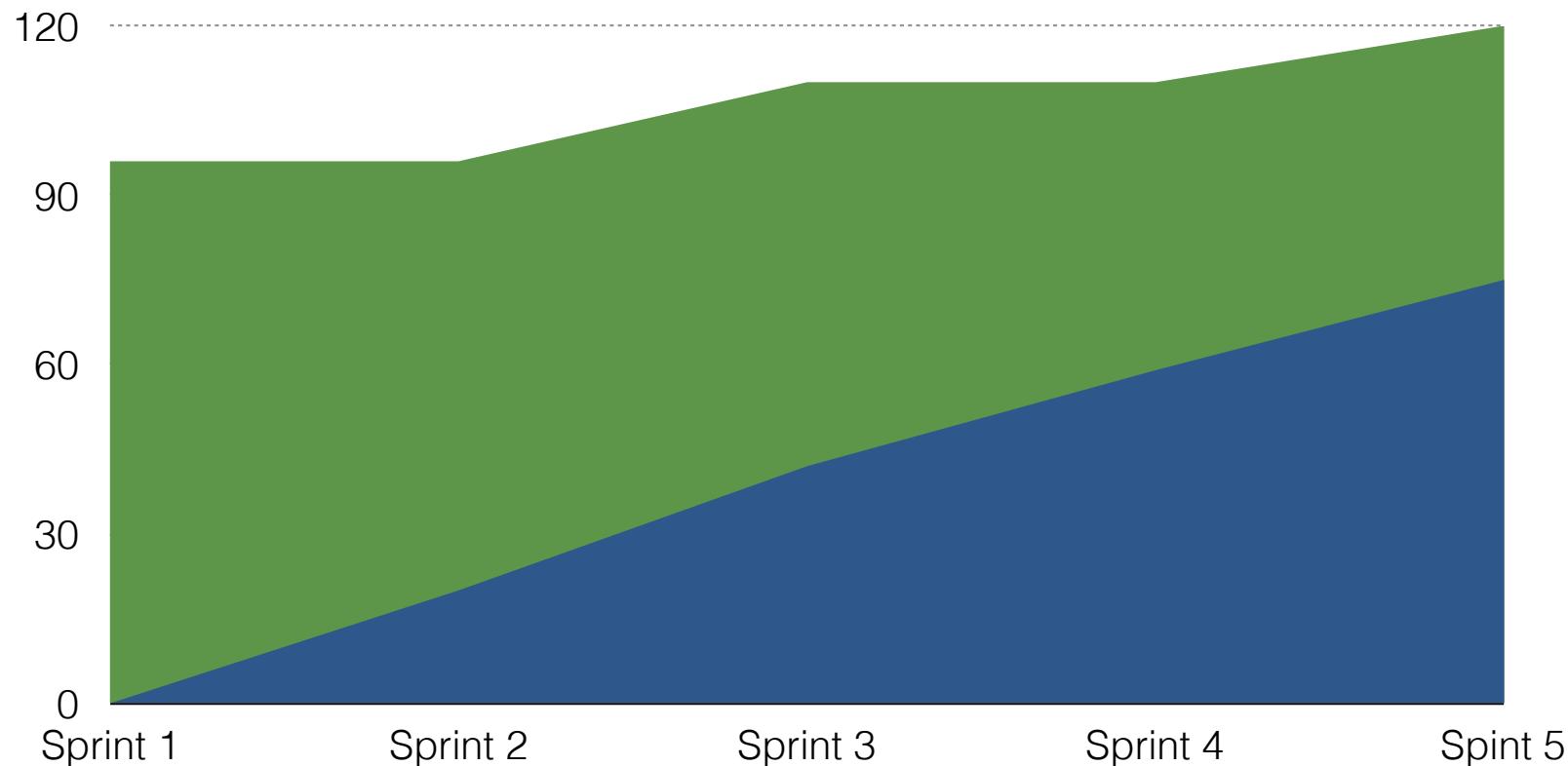
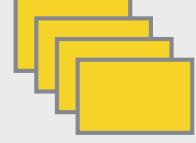
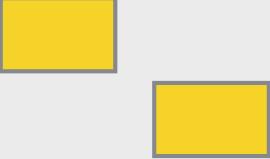
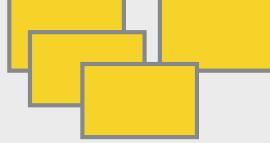


Tableau des stories

Story	To do	In progress	To verify	Done
story A				
story B				
story C				



JIRA Software

Backlog

QUICK FILTERS: Product Recently updated Only my issues Server UI

EPIC: All Issues

Sprint 1: 14 issues

Sprint 2: 6 issues

Start: 10 Aug 2015 — Release: 9 Oct 2015

Issues:

- TIS-25 Engage Jupiter Express for outer solar system travel
- TIS-37 When requesting user details the service should return prior trip info
- TIS-9 After 100,000 requests the SeeSpaceEZ server dies
- TIS-7 500 Error when requesting a reservation
- TIS-10 Bad JSON data coming back from hotel API
- TIS-18 Enable Speedy SpaceCraft as the preferred individual transit provider

Backlog: 49 issues

Issues:

- TIS-25 Engage Jupiter Express for outer solar system travel
- TIS-37 When requesting user details the service should return prior trip info
- TIS-9 After 100,000 requests the SeeSpaceEZ server dies
- TIS-7 500 Error when requesting a reservation

Trello

Project Awesome

Things Coming Up

- Attend Construction Risk conference and share notes with team.
- Research partnership opportunities.
- Organize fundraiser block party.
- Review proposal from Urban Playhouse.
- Call contractors to confirm timeline.
- Give feedback on new site design.

Today

- Secure builder's license.

Done

- Review local zoning rules.

Waiting

- Complete market analysis.
- Hire a contractor for renovations.



Feedback

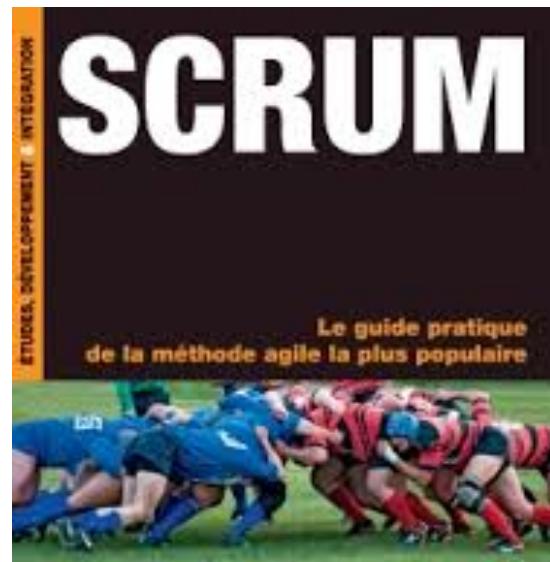
- Date fixée en début de sprint
- Montrer les fonctionnalités **finies** du sprint
- Recueillir les retours des parties prenantes



S'améliorer

- Revenir sur les faits marquants du Sprint
- Analyser les bonnes pratiques et les moins bonnes
- Fixer des objectifs d'amélioration pour chacun

Référence



Claude Aubry

Preface de François Bonnefond

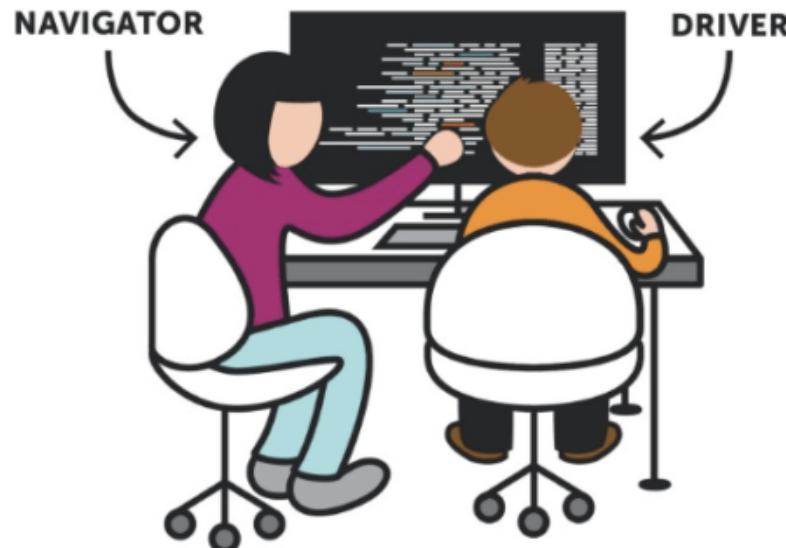
3^e édition



XP a imposé

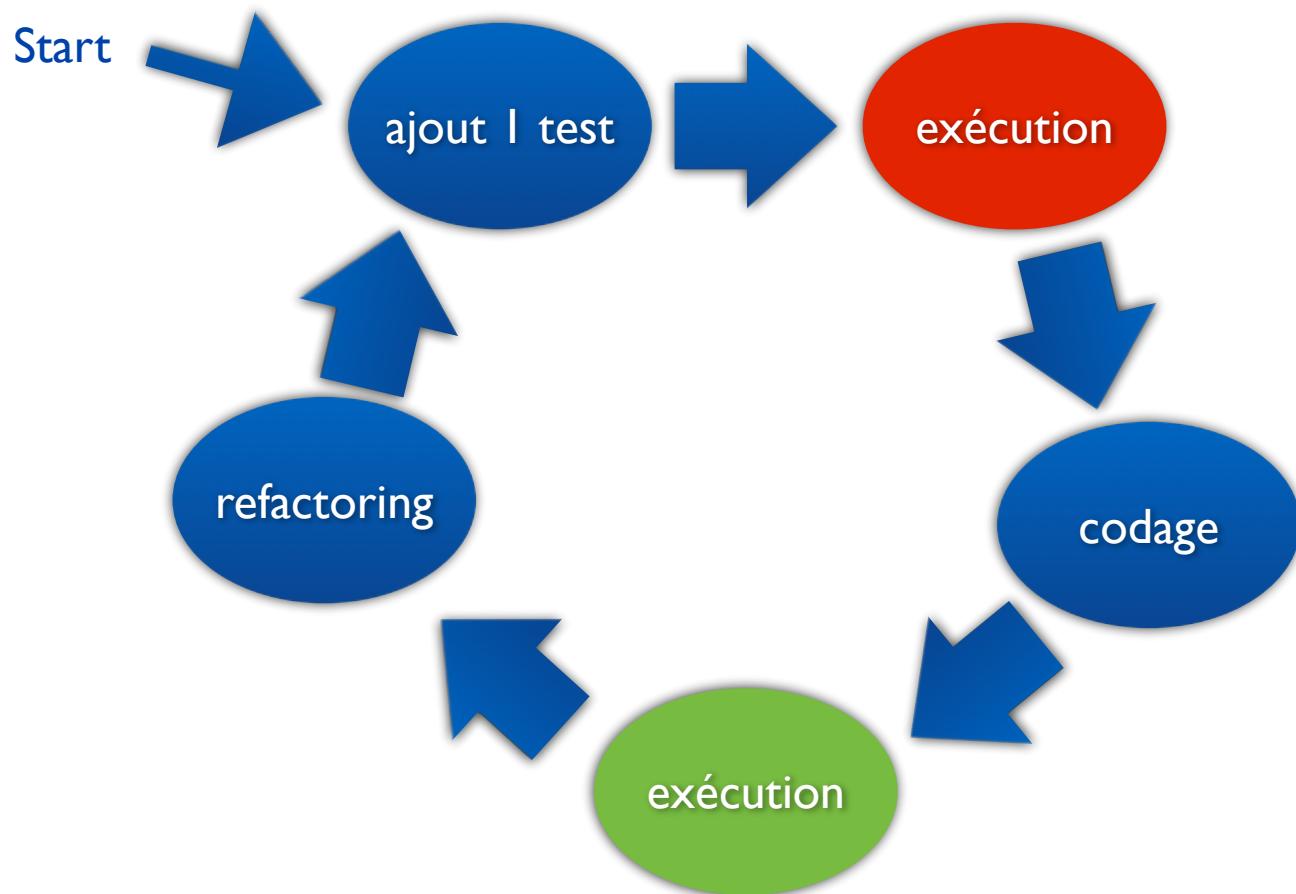
- Pair Programming
- TDD
- Intégration Continue

PAIR PROGRAMMING



- Principe : associer 2 développeurs pour la réalisation d'une tache
- <https://martinfowler.com/articles/on-pair-programming.html#Styles>

Test Driven Development



IC

