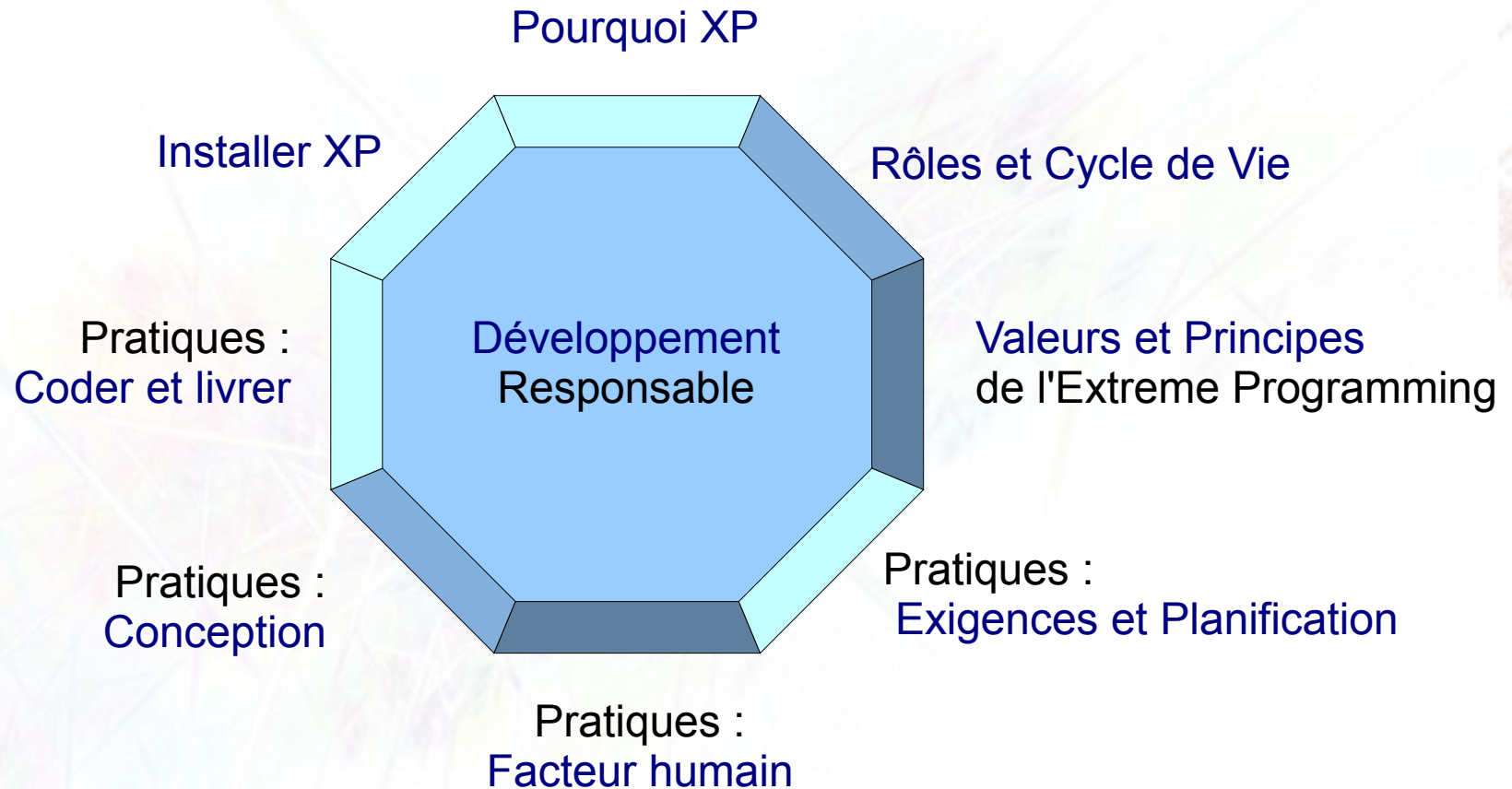


# XP : plus qu'Agile

## Extreme Programming v2 et Développement Responsable

*Thierry Cros*

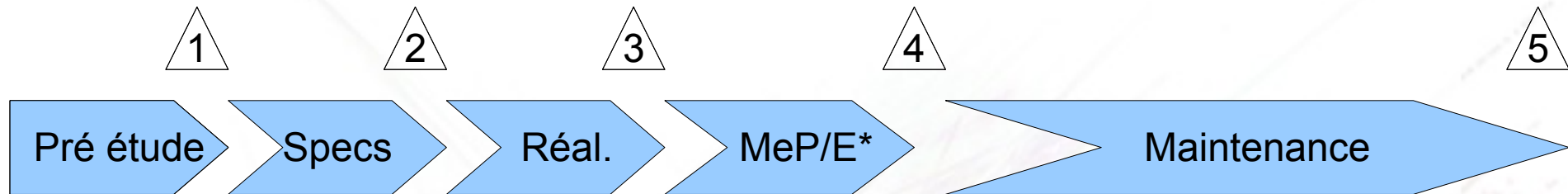
# XP : plus qu'agile



# Pourquoi XP ?

- Pourquoi un « soft » ?
  - Améliorer un process métier
  - « Time to Market »
  - Répondre aux changements
  - Maîtriser les variables « projet »
- Nouveaux matériaux
  - Des cartes perforées...
  - ... à Eclipse
  - Logiciel ou *software* ? ...
  - ... Un matériau souple, malléable : en tirons-nous partie ?
- Projets : 2/3 d'échecs

# Cycle de vie... avant



## MOA

1. Grosso modo c'est cela qu'on veut
2. Là, aujourd'hui c'est cela qu'on veut
3. C'est maintenant qu'il faut tester ?
4. Bon...Enfin on démarre !
5. Ah bon, faut arrêter l'ancienne appli ?

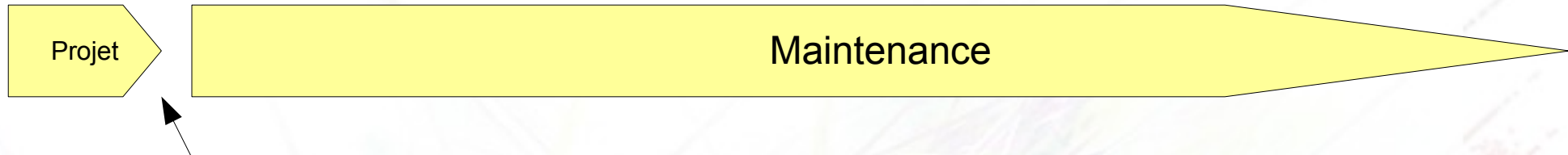
## MOE

- <=> Cela doit être faisable  
*la grosse « vision »*
- <=> C'est cela qu'on a écrit (compris ?)  
*la grosse « spécification »*
- <=> Ca nous rassurerait un peu...  
*les gros documents techniques*
- <=> Bon... Enfin c'est fini !  
*les docs de maintenance*

\* MeP/E : mise en production puis exploitation

# Organisation... avant

Deux équipes



« Au revoir Messieurs Dames... »

Plusieurs collègues, équipes



Utilisateurs



Développeurs



Exploitants

# XP : « programming »

- Quelles activités apportent vraiment une Valeur Ajoutée ?
- Programming
  - « Coder » (Java...)
  - Paramétrer
- Programming = Réaliser le produit

Extreme **Programming** = centré **réalisation**, *ie* **Valeur Ajoutée**

Remettre en cause, réduire voire éliminer les activités « non V.A. »

# XP : « extreme »

- Une fois les activités « non VA » réduites, quelles activités conserver et dans quelle proportion ?
- Dialogues
- Tests
- Conception
- Relecture

**Extreme Programming** = « Pousser à fond » certaines activités

# XP : Extreme Programming

- XP : pousser à fond les activités qui ont prouvé leur utilité.
- Un système **complet** pour le développement de systèmes à forte composante logicielle.
  - Rôles
  - Cycle de Vie
  - Valeurs et Principes
  - Pratiques
    - Gestion des exigences et Planification
    - Facteur humain, Équipe complète
    - Conception
    - Réaliser et Livrer
- XP : plus qu'Agile.

# Retour sur investissement

- XP v1 : « Petit investissement de départ »
- XP v2 : Science de l'économie
  - livrer au plus tôt et régulièrement,
  - réduire le délai entre réalisation et mise à disposition

# Bref historique

- 1999 XPEv1
- 2000 les débuts en France
- Projets « pionnier »
- 2004 XPEv2
- Solution viable

# Rôles

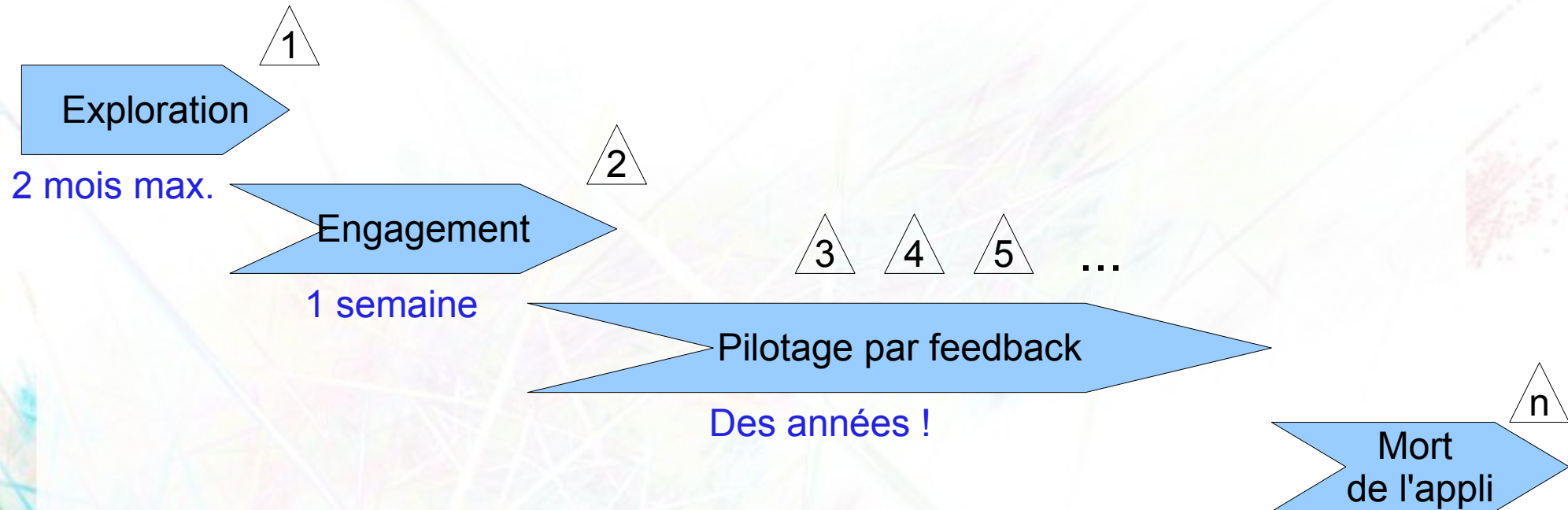


- Une équipe, plusieurs rôles
  - Client  
Spécifie les demandes et les tests-client, planifie en tenant compte de la VA des demandes
  - Développeur  
Estime les demandes, réalise
  - Manager  
Fait confiance, aplanie le terrain,
- Un point focal : la V.A. offerte par le produit

# Autres rôles

- Coach
  - Pose des questions
  - Ne manage pas
  - *Tiens la main de celui qui tient le stylo (le clavier)*
- Tracker
- Testeur
- Consultant occasionnel

# Cycle de Vie XP



1. Fin d'exploration : carottages : architecture, périmètre initial, estimations

2. Engagement : premières valeurs des variables projet :

- Coût
- Délais
- Périmètre
- Qualité

*Principe : Le client en choisit 3, le Développeur la quatrième.*

3. 4. 5. ... Les différentes versions livrées

n. Fin de l'application.

# Architecture

- C'est quoi l'architecture ?
  - Squelette de la solution : framework, matériels et logiciels de base (BDD...)
- Conception émergente : dans le cadre de l'architecture.
- Nous retardons autant que possible les coûts d'infrastructure, ils sont engagés quand ils sont *évidents*.

# Valeurs de l'Extreme Programming

Valeur : norme de conduite personnelle et/ou sociale\*

- **Communication**

Combien d'exemples... et de contre-exemples !

- **Feedback**

Pour contrôler à partir d'éléments les plus objectifs

- **Simplicité**

Le pari de l'Extreme Programming, pour tous, tout le temps

- **Courage**

De changer de rôle, de vision du produit

- **Respect**

Respecter et être respecté en tant que personne

Effet « fractale »



*Ensemble de Mandelbrot*

Mise en œuvre de XP :  
« Conseil constitutionnel »  
qui assure la mise en œuvre des Valeurs.

# XP : les principes

- Humanisme
- Économie
- Bénéfices mutuels
- Autosimilarité
- Amélioration continue
- Diversité
- Réflexion
- Flot continu
- Opportunité
- Redondance
- Échecs
- Qualité
- Petites étapes
- Responsabilité choisie

# Pratiques de l'Extreme Programming

## 13 pratiques de base

Un découpage arbitraire\* :

- Gestion des exigences et planification
- Le facteur humain
- Conception
- Coder et livrer

\* proposée par Michele Marchesi

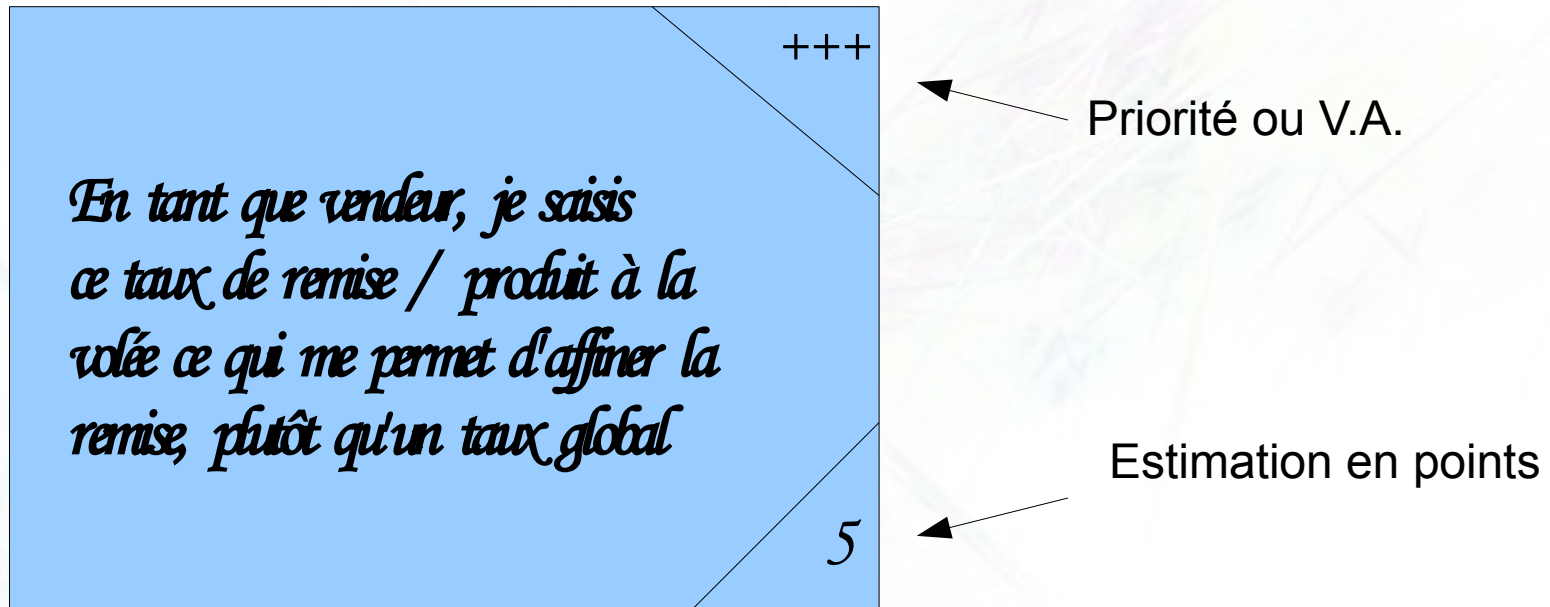
# Histoires et Planification 1/4

- Exigences ? Les besoins sont décrits sous forme d'**histoires** d'utilisation\*.
- Le cycle est itératif incrémental, la durée par défaut d'une itération est d'une **semaine**.
- La vision globale – produit, pratiques – est planifiée à l'échelle du **trimestre**.
- Une **marge de sécurité** – activités optionnelles – permet d'absorber les imprévus.

- Stories  
- Weekly Cycle  
- Quarterly Cycle  
- Slack

\* Ce sont les User Stories : « En tant que... les interactions... pour une V.A .... »

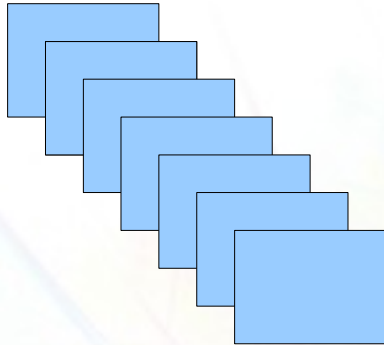
# Histoires



Une histoire est :

- 1) un déclencheur de discussions dans l'équipe
- 2) l'unité de planification (correspondance histoire / itération)

# Itération d'une semaine\*



Ensemble des histoires du produit  
(Backlog)

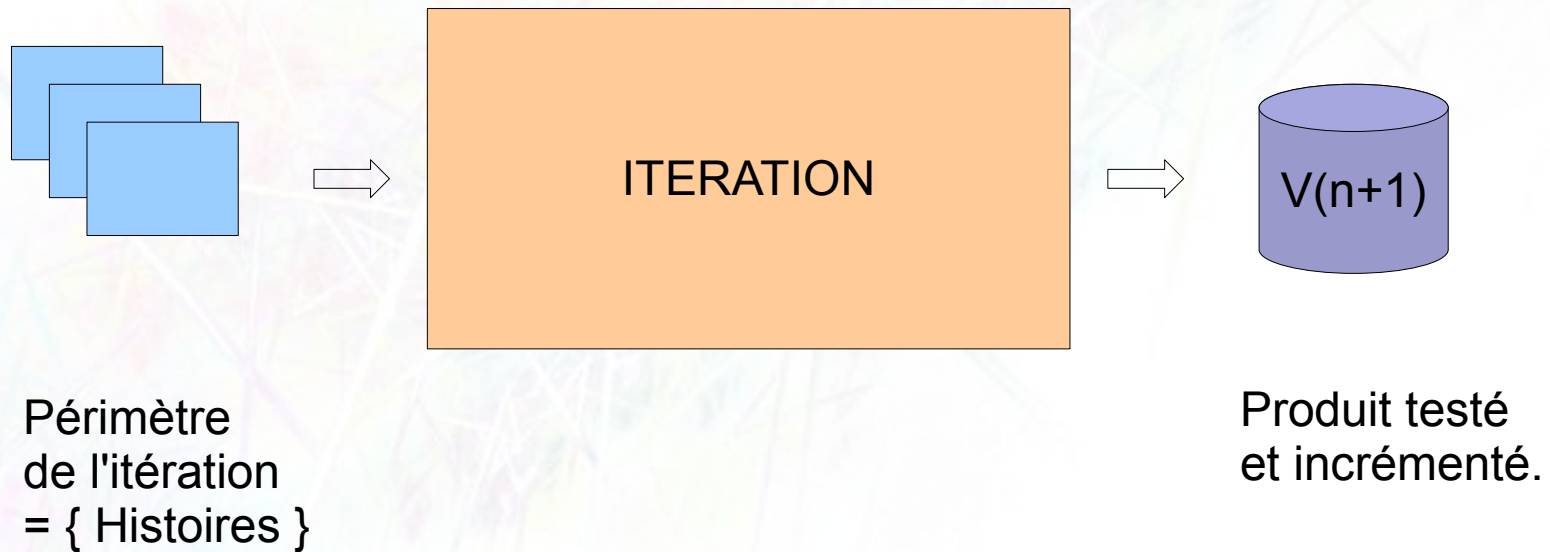

La capacité d'une itération :  
Nombre de cases = vélocité

Tableau non infini !

Le Client planifie en fonction  
1. des priorités des histoires  
2. de leur estimation.

\* En phase d'adaptation, la durée est... adaptée.  
Une semaine reste une bonne « côte », humainement significative.

# Itération, objectif : réaliser les histoires



# Facteur humain, équipe 2/4

- Un plateau-projet, où nous sommes **assis ensemble**
- **Équipe complète** : esprit d'équipe, entr'aide
- **Espace d'information** : les infos sont affichées, disponibles directement
- Un **rythme viable** pour tous les membres de l'équipe : Développeurs, Client...
- Nous codons en **binôme**.

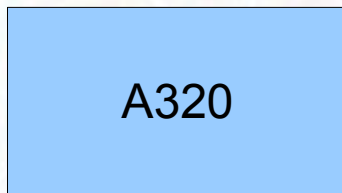
# Conception 3/4

- **Conception émergente** ; une règle d'or, pas de duplication de code.
- **Test Driven Development** : d'abord écrire le test, puis coder, puis mettre au point, période de 1/2 heure.
- A chaque défaut : **analyse des causes** pour éliminer la cause et pas uniquement le défaut

# Conception émergente

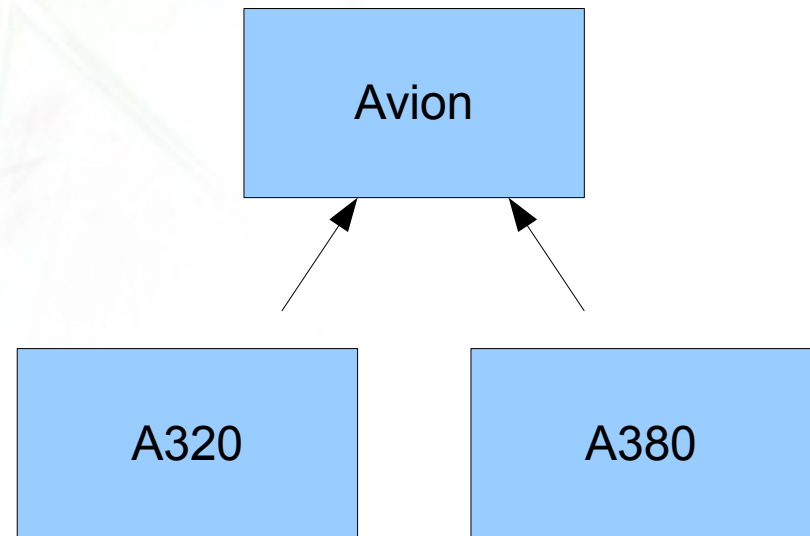
- Itération 1

- Histoires d'utilisation A320



- Itération n

- Histoires du A380



***YAGNI!***

# Coder et livrer 4/4

- Nous testons et intégrons très souvent, le **build** dure **10 minutes** maximum.
- **Intégration continue**, plusieurs fois par jour.
- Le **code-source** et **les tests** sont les documents obligatoires
- Le **code** est **partagé** entre tous
- Gestion de configuration : **une seule version officielle** (qui évolue en permanence)
- **Déploiement au plus tôt** : chaque nuit... chaque deux semaines ou au maximum chaque trimestre

# Installer XP

- Full XP ?
  - Planification / Équipe complète / User stories / Tests-Client
  - + Ingénierie : Conception émergente / TDD
- Quel cadre ?
  - Client : partie prenante, « dans le coup »
  - Projet « perdu »  
ou volonté forte de changement
  - Techniques qui autorisent le refactoring
  - Capacité à changer

# Shu Ha Ri

- Attention à l'impression de « déjà vu »
- Attention aux imitations !
- **Shu**  
Appliquer strictement pour comprendre
- **Ha**  
Jouer avec les pratiques
- **Ri**  
Adapter XP

Une mauvaise idée ? *Ri sans Shu.*

# S'adapter

- La v1 de XP proposait un principe *d'adaptation locale*.
- Ainsi, l'installation de XP s'adapte au contexte...
- ... L'approche agile est une nouvelle vision, conception
  - du projet,
  - du produit logiciel,
  - des rôles.
- Le contexte s'adapte alors à l'Agilité.

# Développement Responsable

- *Imaginez*  
Client, Manager, Développeurs...  
Désormais, vous jouez avec votre argent :
  - En quoi cela change-t-il la perception de votre rôle, de votre responsabilité ?
  - Quelles pratiques conservez-vous ?
- Être assertif pour être responsable.
- Être agile : un savoir-être tout autant que des savoir-faire.

# Merci !

- XP
  - Des rôles
  - Un Cycle de Vie
  - Des Principes
  - Des pratiques
    - Exigences et planification
    - Facteur humain
    - Conception
    - Coder et livrer