

# Agile : XP, Scrum

L'important : être Agile

Thierry Cros – 25.jan.08

<http://agile.thierrycros.net>

1

Cette présentation a pour objectif de mettre en perspective les méthodes XP et Scrum, fers de lance de l'Agile.

# Sommaire

- ◆ [Agile : définition et principes](#)
- ◆ [Scrum](#)
- ◆ [Extreme Programming](#)
- ◆ [Scrum et XP](#)
- ◆ [Devenir Agile](#)
- ◆ [Exemple : « binôme »](#)



# Agile ?



**Manifesto for Agile Software Development**

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions** over processes and tools
- Working software** over comprehensive documentation
- Customer collaboration** over contract negotiation
- Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

<http://agilemanifesto.org/> <http://agile.thierrycros.net> 3

Tout d'abord, il s'agit de revenir à la base : « agile » c'est quoi ?

Je reviens donc au célèbre « Manifeste » de l'Agile, sur le site <http://agilemanifesto.org>

*Nous découvrons de nouvelles façons de développer les logiciels en pratiquant et en aidant les autres à pratiquer.*

*Par ce travail, nous en sommes venus à valoriser :*

***Les personnes et leurs interactions, plutôt que les processus et les outils***

***Un logiciel opérationnel plutôt que des documentations exhaustives***

***La collaboration effective avec le Client, plutôt que les négociations de contrats***

***Répondre et accueillir les changements, plutôt que suivre le plan projet.***

*Autrement dit, alors que nous accordons une certaine valeur aux items de droite, nous donnons plus de valeur à ceux de gauche.*

## Principes « agiles » 1/3

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

<http://agile.thierrycros.net>

4

Ces valeurs « cardinales » de l'Agile se déclinent alors en « principes » plus proches de la réalité du terrain et surtout qui nous permettent de comprendre ce que, concrètement, « agile » signifie.

Je dirais donc que « être Agile » signifie que nous appliquons ces principes, autrement dit, nos méthodes mettent en pratique, déclinent, ces principes « structurants ».

Notez que **la priorité la plus forte est de satisfaire le Client** aux travers de ce que l'on appelle parfois « Versions fréquentes », autrement dit, la livraison de versions incrémentales fréquentes et dès que possible dans le projet.

« Dès que possible » signifie ici : dès que le produit offre une Valeur Ajoutée minimale et suffisante.

Un autre point : **accueillir** les changements.

## Principes « agiles » 2/3

Build projects around motivated individuals.  
Give them the environment and support they need,  
and trust them to get the job done.

The most efficient and effective method of  
conveying information to and within a development  
team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development.  
The sponsors, developers, and users should be able  
to maintain a constant pace indefinitely.



<http://agile.thierrycros.net>

5

On retrouve ici le côté « humanitaire » de l'Agile.

Pour autant, il s'agit de personnes motivées, ce qui fait aussi référence au « développement responsable » de Kent Beck.

Working software : produit opérationnel. Et cela se traduit par des **tests-client** qui permettent de « se forger une opinion ».

## Principes « agiles » 3/3

Continuous attention to technical excellence and good design enhances agility.

Simplicity--the art of maximizing the amount of work not done--is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



<http://agile.thierrycros.net>

6

Simplicité

Auto gestion

Rétrospective : l'équipe s'inscrit dans le « cercle vertueux de la vie ».

## Pourquoi « Agile » ? (2001)

### ◆ Lourd

- Nombreux documents
- Nombreux rôles
- « Contrôle » Qualité
- Suivre les plans
- Jouer pour ne pas perdre
- *Industrie* du logiciel

### ◆ Agile

- Jouer pour gagner
- *Créativité*
- Améliorer la V.A.  
Client != suivre strictement le plan



<http://agile.thierrycros.net>

7

Les méthodes « agiles » étaient dans un premier temps qualifiées de « légères » (light) par opposition aux approches traditionnelles basées sur

- des gros documents
- Des gros plans à suivre
- ...

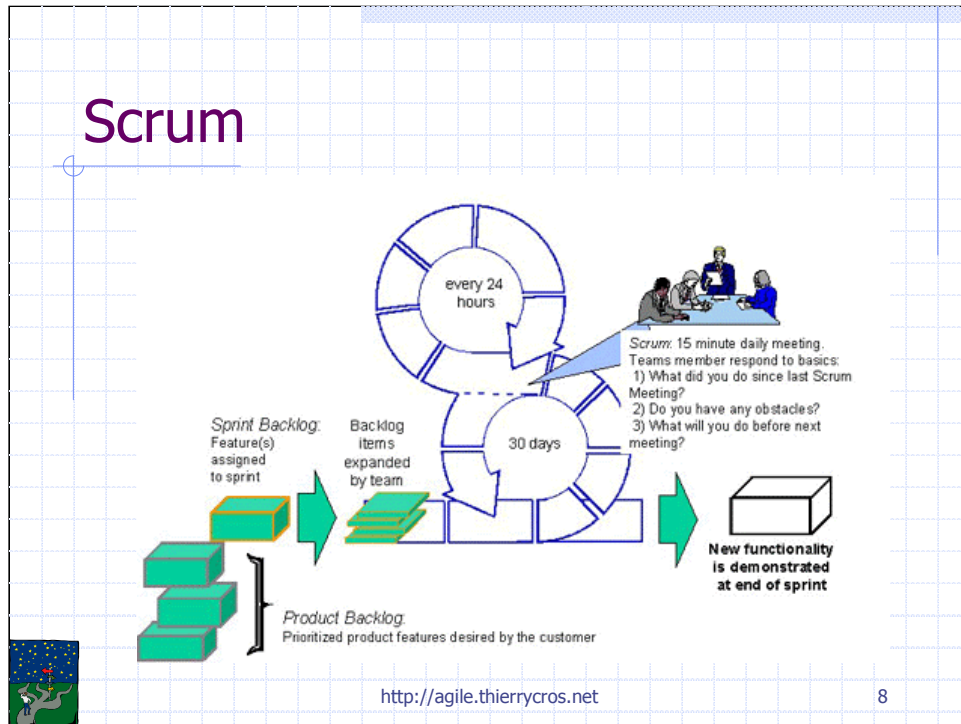
Je crois que ce qui en fait décrit le mieux la différence d'approche est

Jouer pour ne pas perdre

vs

Jouer pour gagner.

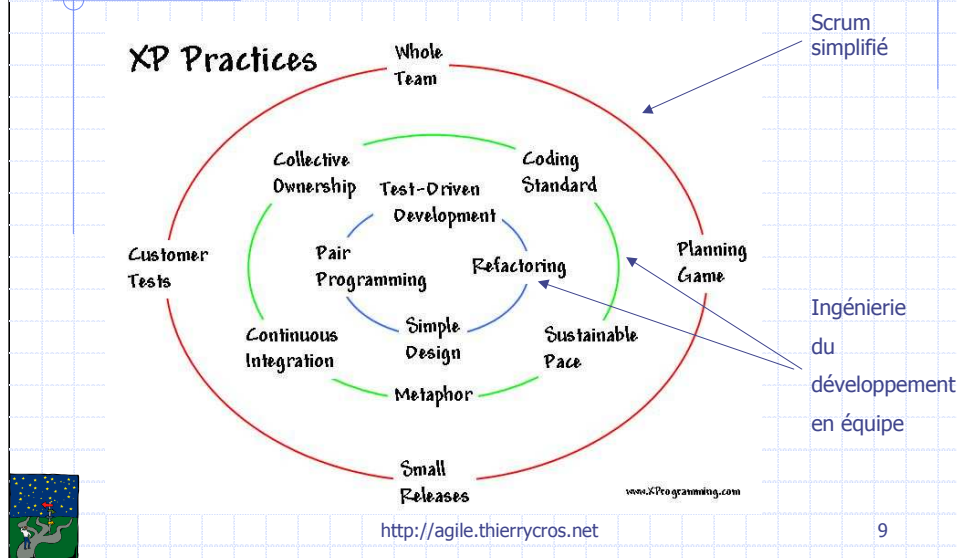
# Scrum



Une méthode également itérative incrémentale, basée sur la V.A. offerte au Client, « sprint » après sprint, autrement dit itération après itération.



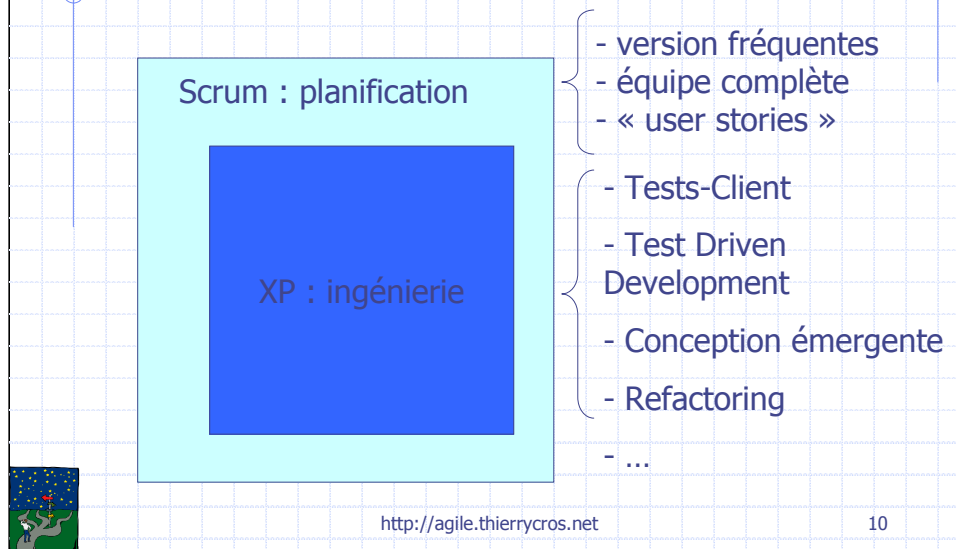
# Extreme Programming



J'aime bien cette présentation de XP en cercle concentriques, qui correspondent à autant de facettes de la méthode.

Le premier cercle est une adaptation de Scrum, les tests-client jouant un rôle important dans XP.

## XP & Scrum



XP et Scrum, un couple gagnant.

En effet, Scrum ne présente pas cet aspect « extrême » de XP tout en s'adressant à l'ensemble des acteurs

**Développeurs**

**Clients**

en particulier.

XP et Scrum est à mon sens un raccourci simplificateur de la réalité. Quid des pratiques XP telles que TDD, refactoring, conception émergente...

## L'important : « être Agile »

- ◆ Une méthode cadre
  - *cf.* Définition de l'Agile
- ◆ Shu Ha Ri
  - Pratiquer la Règle
  - Comprendre la Règle
  - Adapter les pratiques

- Communication  
- Feedback  
- Simplicité  
- Courage  
- Respect

Devenir Agile : gérer le changement vers l'Agile.



<http://agile.thierrycros.net>

11

Au delà des différences entre méthodes, la question est celle de la mise en œuvre d'une méthode agile adaptée à l'organisation.

A mon sens, plusieurs niveaux sont envisageables.

### i) Devenir « plus » agile

Par exemple, gérer plus de rétroactions (feedback), mieux communiquer, créer la notion de « collectif projet », décider de faire plus simple...

### ii) Choisir une méthode

Là, une approche Shu Ha Ri est préconisée : en effet, on a tôt fait de « passer à côté » de la méthode et de son véritable intérêt. D'ailleurs, combien de fois entendons nous par exemple « je fais du XP car je ne rédige pas de doc... » par exemple !

Des pratiques telles que le planning process, TDD, le refactoring, pair programming ne sont pas nécessairement évidentes ; parfois c'est même leur rejet qui semble évident ! D'où l'importance de « pratiquer strictement la Règle » pour en arriver à une compréhension profonde et donc autoriser son adaptation.

## Objectif : Agile

- ◆ État des lieux : symptômes et causes
- ◆ Objectifs et Effets de ces objectifs
- ◆ le Plan adaptatif
  - Qui (client ?...)
  - Quand, projets ?
  - Comment
    - ◆ Faire
    - ◆ Mesurer
  - Combien, budget ?

**Position « meta »**  
Prendre du recul  
et visiter régulièrement  
ces « points de vue »



« Devenir Agile » est un véritable projet.

C'est un projet qui comprend une part significative de changements :

- internes chez les développeurs
- Surtout, au delà, dans la relation au Client, dans les rôles et responsabilités de chacun.

Nous pourrions utiliser la métaphore du voyage. Pour pouvoir établir notre route, mieux vaut savoir où nous en sommes et où nous voulons aller.

D'où cette stratégie en 4 temps :

- i) l'état des lieux : que se passe-t-il actuellement, quels sont les symptômes ou bien désirs qui poussent à devenir agile ?
- ii) Quels sont alors les objectifs poursuivis, pour quel effet, quel bénéfices ? Quelles valeurs ajoutées, pour le Client, le Management, l'équipe, l'organisation.. ?
- iii) Quel est alors le plan pour atteindre cette destination depuis notre point de départ, ce plan étant adaptatif. Quels sont les projets qui peuvent être pilotes ? Quel Client pour participer à ces projets pilotes ? Comment allons nous mesurer nos progrès ? Comment allons nous organiser l'adaptation de ce plan ? Quel budget ?
- iv) Et puis prévoir des « rétrospectives » dans ce projet de passage à l'Agile. Ce que l'on pourrait appeler une « position de recul », une position meta à partir de laquelle nous avons une vue globale de la situation, depuis le point de départ jusqu'à la destination souhaitée.

## Exemple concret : « binôme »

### ◆ Pratiquer la Règle

- Réaliser par binôme
- Développeurs et... non-développeurs
- *e.g.* Architecte (spécialiste) : intègre l'équipe
- Pilote et partenaire
- Cycles

### ◆ Comprendre\* la Règle

- Propriété collective
- Montée en compétences
- Diminution du risque « gourou indispensable »
- Qualité de la production

### ◆ Adapter « binôme »

- Qui (métiers) ?
- Quand ?
- Comment ?

\* Pourquoi ça marche



<http://agile.thierrycros.net>

13

Prenons un exemple concret : la pratique « pair programming » traduite parfois par « binôme ».

Appliquons simplement Shu Ha Ri à cette pratique.

Donc, pratiquons vraiment.

Pratiquons sans préjugés et en appliquant réellement les règles de binôme telles que les rôles de pilote et partenaire, la relation à d'autres pratiques telles que conception émergente...

De là une compréhension concrète, basée sur l'expérience, des avantages de « binôme », voire des inconvénients dans certains cas particuliers

Et enfin, une adaptation qui tient compte de cette compréhension prouvée par l'expérience.