

Christophe Leroy – Marc Lainez

L'Agilité est-elle soluble dans la culture francophone ?



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.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

Equipe

Personnes et interactions plutôt que processus et outils

Résultat

Logiciel opérationnel plutôt que documentation complète

Collaboration

Collaboration avec le client plutôt que négociation de contrat

Changement

Adaptation au changement plutôt que suivi d'un plan

Principles behind the Agile Manifesto

We follow these principles:

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.

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.

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.

Les 12 principes Agiles

1. « Notre première priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à forte valeur ajoutée ».
2. « Le changement est bienvenu, même tardivement dans le développement. Les processus agiles exploitent le changement comme avantage compétitif pour le client ».
3. « Livrer fréquemment un logiciel opérationnel, avec des cycles de quelques semaines à quelques mois, avec une préférence pour les cycles les plus courts ».
4. « Les responsables Métier et les développeurs doivent travailler ensemble quotidiennement tout au long du projet ».
5. « Réalisez les projets avec des personnes motivées. Donnez leur l'environnement et le soutien dont elles ont besoin, et faites leur confiance pour atteindre les objectifs fixés ».
6. « La méthode la plus simple et la plus efficace pour échanger de l'information est une conversation en face à face ».
7. « Un logiciel opérationnel est la meilleure mesure de la progression du projet ».
8. « Les processus agiles encouragent un rythme de développement soutenable. Commanditaires, développeurs et utilisateurs doivent pouvoir maintenir indéfiniment un rythme constant ».
9. « Une attention continue à l'excellence technique et à la qualité de la conception renforce l'agilité ».
- 10.« La simplicité - l'art de maximiser la quantité de travail à ne pas faire - est essentielle ».
- 11.« Les meilleures architectures, spécifications et conceptions sont issues d'équipes qui s'auto-organisent ».
- 12.« À intervalle régulier, l'équipe réfléchit aux moyens de devenir plus efficace, puis ajuste son mode de fonctionnement dans ce sens ».

La croyance en l'homme providentiel ...

ou au « deus ex machina » ...

... « Nous vivons sous un prince ennemi de la fraude. »

(Tartuffe, acte V, scène VII)

... ou au Père Noël ...

Le beau jeu (en équipe) ...

- Moi, je suis pour l'Agilité ... mais c'est moi qui décide

- De toute façon, je l'avais bien dit ...

- C'est pas moi, c'est l'autre !

La croyance en la méthode providentielle

« Si on se met à l'Agile, tous nos problèmes actuels vont se résoudre ! »

... *Il n'y a pas de recette miracle*

2 – Logiciel « qui marche » plutôt que documentation complète

Ah l'administratif !

« avant de débiter les développements, il faut absolument que tu aies rempli le formulaire ABC123 et que tu aies l'accord officiel des services X, Y et Z »

Le plan projet n'est pas le projet (cf. la pipe)

Le plan projet est OK et indique que la fonctionnalité « à votre bon cœur » a été validée en tests de Recette ; nous avons juste un (petit) souci de performance qui empêche toute montée en production



La trahison des images. René Magritte. 1929
huile sur toile, 59 x 65 cm
Los Angeles County Museum

Nom de la tâche	Contrainte	Travail	Durée	Début	Fin
Projet Market 1	Dès Que Possible	290,7 jours	278 jours?	Lun 05/09/05	Lun 31/10/05
Lancement	Dès Que Possible	11,5 jours	18 jours	Lun 26/09/05	Mer 11/10/05
ASE Point d'étape initiale	Doit Commencer Le	0 jour	2 jours	Mer 18/10/05	Mer 11/10/05
Phase 1	Dès Que Possible	279,2 jours	258 jours?	Lun 05/09/05	Lun 01/11/05
Conception	Dès Que Possible	279,2 jours	88 jours?	Lun 05/09/05	Lun 01/11/05
JE-CONCEPTION	Début Au Plus Tôt L	0 jour	0 jour	Mer 02/11/05	Mer 02/11/05
CHANTIER TECHNIQUE	Dès Que Possible	0 jour	22 jours	Lun 05/09/05	Mer 04/10/05
CHANTIER SOLUTION	Dès Que Possible	268,2 jours	65 jours?	Lun 26/09/05	Lun 01/11/05
Planning détaillé de la conception	Dès Que Possible	11,5 jours	5 jours	Lun 26/09/05	Ven 31/10/05
Conception Fonctionnelle	Dès Que Possible	200,7 jours	40 jours?	Lun 26/09/05	Lun 27/10/05
Ventes Distribution	Dès Que Possible	66,6 jours	34,5 jours?	Lun 26/09/05	Lun 14/10/05
Ateliers & préparation	Dès Que Possible	39 jours	34,5 jours	Lun 26/09/05	Lun 14/10/05
Données structurantes - 1 Ventes et Distribution	Début Au Plus Tôt L	1 jour	0,5 jour	Jeu 03/11/05	Jeu 03/11/05
Données structurantes - 2 Ventes et Distribution	Début Au Plus Tôt L	1 jour	0,5 jour	Lun 07/11/05	Lun 07/11/05
Données article vente - 1	Début Au Plus Tôt L	0,5 jour	0,5 jour	Mer 08/11/05	Mer 08/11/05
Données article vente - 2, nomenclatures	Début Au Plus Tôt L	0,5 jour	0,5 jour	Lun 14/11/05	Lun 14/11/05

Ceci n'est pas le projet.

3 – Collaboration avec le client plutôt que négociation de contrat

Le consensus, ... vous n'y pensez pas !

“Un bon rapport de force vaut mieux qu'un mauvais consensus”

Comment traduit-on gagnant-gagnant en français ?

« Je vous ai compris »

Message du sponsor projet en conclusion du kick-off :

« Ce magnifique projet au forfait dont la mise en production est planifiée le 1^{er} avril de l'année n+2, qui couvre un ensemble de 179 fonctionnalités contractuellement identifiées et qui s'inscrit dans un paysage système de 10 applications, ne pourra être qu'un retentissant succès ! »

4 – Réagir au changement plutôt que suivre un plan

C'est pas le bon service ...

« ... adressez-vous ... ailleurs ... »

Il est important de suivre la Vision initiale du projet

Si on sort de cette Vision, on ne développe plus le même projet
(Business Value Model, Document de Vision, ...)

Cela ne veut pas dire qu'il n'y a pas de plan

« Ca ne sert à rien de faire un plan, on fait de l'Agile... »

Courage ... fuyons !

Extrait du CoPil du xx/xx/xxxx :

Les dates limites sont dépassées.

La durée des tests Métier sera donc réduite afin de garantir le démarrage du projet dans le respect du planning ...

Il faut mettre en place les outils permettant de le faire

Planning en user stories "full-stack" (fonctionnalité métier)

Remettre en cause les processus de développement

Développement piloté par les tests

2- Le changement est bienvenu, même tardivement dans le développement. Les processus agiles exploitent le changement comme avantage compétitif pour le client

Flexibilité, ...

Le planning projet a été validé lors du dernier CoPil, je compte sur vous pour le suivre à la lettre

Vous avez dit « open » ?

Je vous rappelle que vous avez officiellement validé les spécifications il y a 1 an. Maintenant que nous venons de livrer les développements associés, il est hors de question de modifier quoique ce soit ... *un petit avenant, peut-être ?*

Ce n'est pas non plus « open bar »

A chaque itération, le plan doit être mis à jour

3- Livrer fréquemment une application fonctionnelle, toutes les deux semaines à deux mois, avec une préférence pour la période la plus courte

« Acceptance tests »

Cri d'effroi d'un utilisateur « volontaire » au début des tests de validation
Métier :

« Mais ils n'ont rien compris au métier ! Jamais nos équipes ne pourront travailler avec cet outil. »

Le client finit parfois par en être lui-même demandeur

« ... donc tout ça, c'est fini ? On peut mettre en prod cette semaine ? »

Je ne sais pas ce que je veux ...

... mais je suis contre les idées des autres

ou la difficile mission d'être un Product Owner dans la douce France ...

Je sais ce que je veux ... mais ce n'est pas ce dont j'ai besoin ...

C'est en comprenant les problèmes du Métier qu'on trouve les meilleures solutions, tous ensemble

MOA / MOE

Qui peut traduire MOA / MOE en anglais ? ... en allemand ?

Moi pas parler Informatique ... Moi pas parler Métier ...

« Moi, je suis pour la communication ...

mais c'est l'autre qui apprend ma langue ... »

5- Bâissez le projet autour de personnes motivées. Donnez leur l'environnement et le soutien dont elles ont besoin, et croyez en leur capacité à faire le travail

Faites ce que je dis ... (ou les respect des valeurs)

- On va faire le Sprint Planning la semaine prochaine et Jean Dupont définira les charges ...
- Bon Sprint Planning, et n'oubliez pas qu'on met en prod le xx/xx/xxxx !
- Et vous ferez comme je vous ai dit ...

La bonne personne au bon ...

« Nous avons besoin d'un contact Métier pour leur projet informatique. Tu as quelqu'un qui n'a rien à faire actuellement ? »

Le droit à la formation ... mythe ou réalité ?

Moi je veux le programmeur de 22 ans avec 10 ans d'expérience, qui connaît toutes les nouvelles technologies, qui parle anglais, qui travaille dur, qui rédige bien, ... et qui sera toujours au top dans 40 ans ...

... ah, j'oubliais : et qui ne coûte pas cher ...

La paperasse ...

Et un petit rapport en 20 exemplaires !

Ah, le charme des réunions à la française ...

- Arriver à l'heure !? Et puis quoi encore !
- Moi, je n'ai rien à dire ... mais j'aime bien parler
- Moi, je sais tout ... mais je ne dirai rien
- Moi, je n'ai pas compris ... mais je n'ose pas le dire
- ...
- Mais qu'est-ce que je fais là ?
- ...

Ne vois-tu rien venir ?

D'après nos indicateurs, le budget du projet est bien en ligne avec le plan initial. Nous avons consommé toutes les charges de réalisation prévues, il ne reste plus qu'à commencer les tests ...

8- Les processus agiles promeuvent un rythme de développement soutenable. Sponsors, développeurs et utilisateurs doivent pouvoir maintenir le rythme indéfiniment

On va tout donner !!!

Un bon petit coup de ... collier, ça n'a jamais fait de mal à personne !

Le désert des Tartares

Soupir d'un développeur :

"Çà fait 3 semaines que j'attends la validation des spéc's pour commencer le développement ..."

L'angoisse

Soupir de l'équipe Exploitation :

"La mise en production est dans 2 semaines et nous n'avons toujours pas testé la séquence globale des batches ... *ça nous promet de belles nuits*"

Les rois du système D

« L'aventurier aime la discipline »

Pierre Mac Orlan

Après moi ...

Je n'ai pas tout compris à quoi ça sert mais, bon, on verra bien ...

Eviter d'aggraver la dette technique

Mettre en place des outils de refactoring

Utiliser la méthode Mikado

Pourquoi faire simple quand on peut faire compliqué
... une cathédrale magistrale plutôt qu'un bazar peinard ...

Propos recueillis dans le Thalys Paris-Bruxelles

- Moi, j'ai vu que chez XXX, ils utilisaient cette fonctionnalité : on devrait quand même la prendre !?

Mais que fait le gouvernement ?

"Ask not what your country can do for you, but what you can do for your country."

John F. Kennedy

De l'esprit des lois

« C'est une expérience éternelle que tout homme qui a du pouvoir est porté à en abuser (...) Pour qu'on ne puisse abuser du pouvoir, il faut que, par la disposition des choses, le pouvoir arrête le pouvoir. »

Montesquieu

12- À intervalle régulier, l'équipe réfléchit aux moyens de devenir plus efficace, puis ajuste son mode de fonctionnement dans ce sens

Le droit à l'erreur ... en Version Française

L'équipe, l'équipe, l'équipe ... mais si tu te trompes, t'es viré

« La société de défiance :

Comment le modèle social français s'autodétruit »

Yann Algan, Pierre Cahuc

Editions Rue d'Ulm, 2007

Trust

Soyons efficaces !

« Bon, pas de temps à perdre en discussions, on retourne à la mine »

Les rois de la critique

Mais quid de l'auto critique ?

... et de l'esprit critique ?

et pour finir, le fameux « Ça va pas le faire ! »

« Ce n'est pas parce que les choses sont difficiles que nous n'osons pas,
c'est parce que nous n'osons pas qu'elles sont difficiles »
Sénèque

***Bon vent dans votre
aventure Agile !***

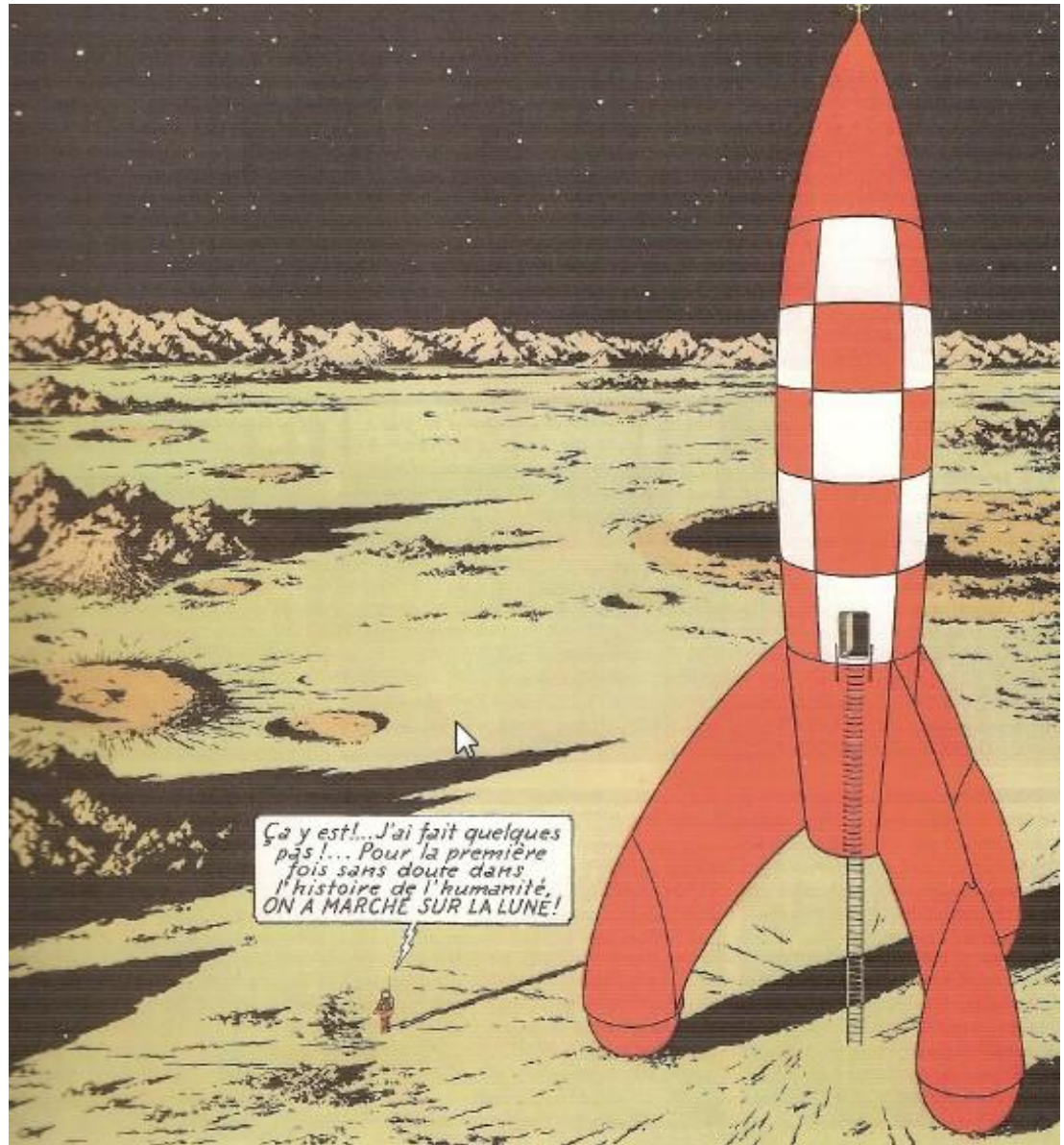
... que le ciel ne nous tombe pas sur la tête ...



« Astérix et le chaudron »
Uderzo & Goscinny, Dargaud Editeur

... et nous décrocherons la lune ...

« On a marché sur la lune »
Hergé, Casterman Editeur



L'Agilité est-elle soluble dans la culture francophone ?

Christophe Leroy

christophe.leroy@atipyc.com

+33 (0)6 86 28 31 29

www.atipyc.com

Marc Lainez

marc.lainez@gmail.com

+32 474 78 66 33

[@mlainez](#)

ANNEXES

1. Eliminer les gaspillages
2. Favoriser la connaissance
3. Décider le plus tard possible
4. Livrer rapidement
5. Donner le pouvoir à l'équipe
6. Penser qualité dans la construction du système
7. Optimiser l'ensemble

Communication

Simplicité

Feedback

Courage

(Respect)

Confiance

Tout projet repose sur la confiance ...

Confiance entre les équipes Métier et Informatique, confiance entre les interlocuteurs internes et externes, ..., confiance en la réussite du projet, ...

Seule la confiance partagée autorise la prise de risque inhérente à tout projet et les nécessaires conflits générateurs de progrès.

Cette confiance n'est pas un don du ciel : elle s'appuie sur un respect, une simplicité, un engagement et un haut niveau d'exigence entre tous les acteurs du projet.

(Une bonne dose d'humour ne fait pas de mal non plus !)

Communication

Tout projet est l'apprentissage d'un nouveau langage ...

Parlez-vous SAP ? Parlez-vous e-procurement ? Parlez-vous Web services ?

Parlez-vous segmentation client ? ...

Tout projet s'apparente à la construction d'une tour de Babel ... inutile d'espérer atteindre le ciel si les différents corps de métier ne se comprennent pas.

Cette communication requiert une profonde humilité pour apprendre la langue des autres et une grande honnêteté pour reconnaître sa non compréhension, le cas échéant.

Changement

Tout projet est changement ...

Formation, Change Management, ... certes !

Mais quid de l'expérimentation, du droit à l'erreur, de la formation « sur le tas » (ou du « Pair Programming » !?), de la montée en compétence par métissage des équipes ?

Une attitude positive face au changement se bâtit tout au long d'un projet par un enchaînement de réalisations de plus en plus conséquentes : « avant de décrocher la lune, grimpons le Mont Blanc ! »