

Réussir un projet plein d'enjeux grâce à des méthodes agiles bien utilisées

Ian Stokes, janvier 2016

Le « manifeste agile »
une déclaration pour l'agilité

Des individus et des interactions plutôt que
des processus et des outils



La vraie coopération
plutôt que le recours
contractuel



La réactivité face aux changements plutôt
que le suivi d'un plan



Des résultats qui donnent satisfaction plutôt qu'une
documentation exhaustive

Un autre perspective sur le management d'un projet agile

1) Introduction

Pour toute activité économique évoluant dans un contexte d'entreprise et ayant en jeu des moyens et des processus organisationnels, des aspects tels que le climat concurrentiel et les opportunités technologiques sont évolution constantes. Une approche agile à la gestion de projet permet à la fois de faciliter et de renforcer l'interaction entre les besoins des clients et le potentiel de la technologie, afin d'évaluer en permanence la valeur du projet et de faire en sorte que l'effort en vaille la peine.

En essence, un projet agile est un projet d'apprentissage. Cette réalité impose des contraintes au projet telle que la nécessité d'expérimenter et de communiquer en utilisant des modèles et des prototypes, d'explorer des options en adoptant différents points de vue, inclure des ajustements bénéfiques sans compromettre la fonctionnalité globale, et de démontrer la maîtrise de l'évolution du champ d'application. Les décisions sont justifiables en cas de valeur ajoutée suffisante par rapport aux alternatives. *

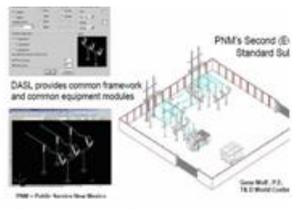
Une conséquence importante de cette logique est que l'appréciation des objectifs métiers devient la base de la coopération, créant ainsi une référence partagée qui est adoptée pour évaluer en permanence l'adaptation aux besoins. Cette approche contraste avec la méthode classique qui consiste à déléguer une spécification bien définie ensuite utilisée comme base pour des tests d'acceptation. Un 'business case' bien développé est encore plus important pour un projet agile que l'habitude, parce que les décisions qui porte sur le périmètre sont faits tout au long du cycle de vie du projet et sont entraînés par une interprétation évolutive des priorités et de l'effort, plutôt que d'un accord contractuel au début de le projet

Ce type d'évaluation d'entreprise nécessite l'engagement, la confiance et les connaissances des différentes catégories de clients, utilisateurs, experts, gestionnaires, concepteurs, constructeurs, opérateurs et investisseurs. Inévitablement avec un tel éventail de contributions des divergences et des conflits d'intérêts apparaissent. Une compréhension relationnelle, la négociation, la résolution créative de problèmes et les fréquentes périodes de turbulences sont abordées avec détermination en adoptant un style de facilitation de l'équipe projet qui repose sur des décisions ouvertes et transparentes et la mise en œuvre d'ateliers de travail en équipe.

Les méthodes agiles de gestion de projets ne se limitent pas aux projets informatiques, bien qu'ils profitent de la capacité à livrer des segments de la solution par incréments testables. Soit fonction par fonction (tranches verticales), comme une nouvelle version qui va élargir le champ d'application à chaque étape de son évolution. Ou bien comme un ensemble que l'on développe progressivement (tranches horizontales), que vous pouvez voir évoluer à chaque étape depuis la première ébauche, à travers la mise en page, la maquette, des simulations, des modèles et des échantillons, des prototypes, et enfin des versions, jusqu'à sa stabilisation dans le produit fini, ou plus probablement un mélange des deux.



1. Site proposé



2. Dessin de l'architecte



3. Image informatique



4. Réalisation complétée

* Lors d'un projet agile, les avantages apportés peuvent évolués tant que les échéances sont respectées, tandis que le périmètre doit se rétrécir au fur et à mesure que les demandes non essentielles sont supprimées au profit des résultats nécessaires.

Un autre perspective sur le management d'un projet agile

2) Le 'business case' d'un projet agile

Un projet de développement agile doit avoir une analyse de rentabilité claire. Le 'business case' décrit toutes les raisons pour lesquelles le projet est nécessaire et souhaitable. Le succès du projet est évalué sur la base des objectifs mesurables qui sont exprimés dans ce 'business case's.

Ainsi, en plus des exigences fonctionnelles et techniques, dans un projet agile les besoins de l'entreprise devraient être bien exprimés. Ils doivent dépasser le périmètre des mesures financières et inclure des résultats non-financières essentiels à la réussite.

Dans un projet agile, tandis que les besoins fonctionnels sont exprimés selon la perspective de l'utilisateur, les besoins globaux incluent des explications adéquates pour justifier le pourquoi du projet et la logique des besoins fonctionnels.

Par exemple:

| <i>Besoin fonctionnel</i> | <i>Besoin technique</i> | <i>Besoin métier</i> |
|---|---|--|
| L'exploitant doit être en mesure de mettre le système sous et hors tension | Le système disposera d'un interrupteur ou un bouton pour l'allumer et l'éteindre | L'opérateur pourra mettre le système sous et hors tension afin de le rendre disponible pour les calculs, l'impression, la mise à jour des données, et pour améliorer la sécurité |
| Le vendeur doit être en mesure d'ajouter un nouveau contact | Le système aura une capacité de saisie de données pour de nouveaux contacts | Le vendeur peut ajouter de nouveaux contacts afin de suivre et de gérer les clients prospectifs |
| Le directeur des ventes doit être capable de rechercher dans la base de données client en utilisant les mots clés | Le système cherchera dans toutes les tables de données relatives aux clients avec des mots clés | Le directeur des ventes peut accéder et utiliser les données relatives aux clients de tous les segments commerciaux afin de comprendre et de superviser le cycle de vente |

Les besoins globaux exprimés au niveau d'entreprise sont importants, car ils servent de référence pour mesurer la réalisation de l'objectif d'un projet géré avec des méthodes agiles. Les besoins globaux, ainsi que les besoins fonctionnels définis à partir d'un point de vue utilisateur représentant la base d'un projet agile.

La clarification des objectifs de l'entreprise n'est pas évidente, car cela signifie qu'il existe un alignement des projets avec la stratégie globale de l'entreprise, la prise en compte de la mission (ce que l'organisation est) et à la vision (ce qu'elle entend être.)

Construire une vision nécessite un esprit fort d'aspiration, d'inspiration et de connaissances des efforts à concerner. La vision crée un narratif, une sorte d'histoire partagée, à partir de laquelle on aura un tremplin pour entendre les récits d'utilisateurs.

Avec cet alignement à la stratégie global, les besoins peuvent être liés à la valeur ajoutée et tenir compte de l'importance et l'urgence du projet. E même temps les coûts sont liés à la complexité et les risques. De cette façon, il y a un potentiel soutenu pour examiner les priorités en termes de valeur par rapport aux coûts.

Un autre perspective sur le management d'un projet agile

Les rôles dans un projet agile sont équilibrés entre les représentants du métier et les utilisateurs d'une part et l'équipe de développement de l'autre. Dans cette optique, les méthodes de gestion d'un projet agile suggèrent une diversité de dispositifs du très simple – représentant métier, chef de projet et équipe de projet - au plus élaboré.

Des rôles comme l'utilisateur ambassadeur (mandaté par les utilisateurs pour représenter leur point de vue), le visionnaire (centré sur les idées et la convergence entre la technologie et l'entreprise), le coordinateur technique (s'occupe de la gestion du contenu, une activité agile fondamentale), l'animateur de l'atelier, ainsi que le chef de projet et le promoteur du projet, peuvent aider à clarifier et à donner une structure organisationnelle à un projet agile.

En pratique, ce dialogue évolutif entre la technologie et le marché implique un échange fréquent entre développeurs et utilisateurs à l'aide d'inspections, de tests, de modélisation, de prototypage et de simulation. Ces événements sont souvent organisés sous la forme d'ateliers et ils sont l'essence de l'approche agile.

Parce que les utilisateurs participent aux tests, autant que possible, et utilisent leurs données réelles dans un contexte authentique, ils sont en mesure d'approuver les produits développés au fur et à mesure de l'avancement du projet. Cela donne un sentiment d'appropriation aux utilisateurs, ce qui est appréciable lorsque le produit doit être approuvé. En fait, l'approbation a lieu systématiquement dès que des versions intermédiaires sont fournies.

Cette démarche d'intégration, de synchronisation constante et d'approbation est rassurante à la fois aux développeurs et aux métiers, parce que des résultats tangibles et visibles se dégagent. Toutefois, elle doit être gérée, parce que la communication peut supprimer ou ajouter des besoins.

La capacité d'évolution et de gestion des changements renforcent les avantages apportés sans pour autant pénaliser la stabilité, parce que la démarche est facilitée par des itérations rapides. Lorsque les livraisons sont fréquentes, il y a toujours une version récente et opérationnelle de disponible, et ainsi les erreurs et les malentendus sont détectés au plus vite.

La philosophie de «l'échec rapide» a pour objectif la détection des défaillances avant qu'elles ne deviennent trop coûteuses. Des projets agiles sont des projets d'apprentissage. Il est préférable de savoir rapidement si une solution ne peut pas satisfaire. Puisque le client est impliqué, les risques d'incompréhension devraient être réduits.

Un projet efficace et agile emploie souvent une technologie basée sur une architecture flexible, modulaire et orientée objet, en utilisant des protocoles communs et des modèles de design qui protègent contre les impacts des changements ou de problèmes de réversibilité.



Business Case



User Story

Un autre perspective sur le management d'un projet agile

3) La planification d'un projet agile

La logique de planification d'un projet agile commence avec une date limite et fonctionne sur le principe de la timeboxe. Une timeboxe a une durée fixe et un contenu limité. L'objectif est de fournir les informations et les fonctionnalités les plus importantes à l'intérieur de la timeboxe et de laisser les éléments les moins prioritaires pour une timebox plus tard.

Le mot « timeboxe » vient du monde du journalisme, car à chaque jour ou chaque semaine dans la vie d'un journal on doit produire des résultats les plus utiles et de valeur ajoutée la plus importante qui soit possible, et cela dans un créneau de temps qui ne peut pas être élargi, et le restant est laissé pour le lendemain. Un autre mot pour une timeboxe est un « sprint », qui signifie une accélération d'activité afin de produire des résultats concrets.

Au niveau global des programmes et du portefeuille de projets, il doit exister un calendrier directeur et un calendrier d'événements métiers. Cependant, la plupart du travail de planification dans un projet agile s'effectue au début de chaque timeboxe.

Le « jeu de la planification » consiste d'une séance de planification réalisée par l'équipe où les tâches définies pour répondre aux besoins sont inscrites sur des cartes. L'effort est évalué en termes de bénéfices, de coûts et de risques, et lorsque le projet avance, les besoins sont à nouveau priorisés et peuvent faire l'objet d'un ajustement des priorités au début de chaque timeboxe.

Les priorités sont souvent basées sur les critères MOSCOW ('must', 'should', 'could', 'wont'), et en tenant dûment compte des contraintes techniques, les tâches sont ordonnées et le temps de travail affecté.

Les besoins 'M' sans ceux que l'on doit réaliser et qui ne sont pas négociables. Ils sont impératifs à la réussite du projet. Sans réponses à ces besoins, la solution serait inopérante, inutilisable, non utilisées ou inutiles, et même dangereuse. Les «must have» sont des exigences contractuelles en règle générale, mais dans ce cas ils devraient être évalués rigoureusement afin de comprendre l'effort nécessaire. Une équipe expérimentée cible typiquement une proportion de deux tiers des «must have » des exigences, par rapport à un tiers pour une équipe agile peu expérimentée.

Des besoins 'S' sont ceux qui « devrait » être satisfaits et peuvent représenter entre un tiers à deux tiers du restant. Une absence de ces besoins serait ennuyeuse, mais ils ne sont pas indispensables à la réussite du projet, car un alternatif serait acceptable.

Des besoins 'C' qui « pourrait » être satisfaits sont ceux qui sont sensibles au compromis. Si une solution à ces besoins n'est pas rendue à l'heure, il y aurait d'autres compensations importantes telles que le respect des dates butoirs, ou d'autres gains tels qu'une fonctionnalité de substitution.

Des besoins 'W' qui «ne seront pas » satisfaits sont soit repoussés à une date future ou bien considérés comme inutiles et ne valant pas l'investissement en énergie.



Besoins priorités



Timebox

Un autre perspective sur le management d'un projet agile

Des mises à jour d'un projet agile doivent être fréquentes, car le tempo d'un tel projet dynamique est très élevé. L'une des raisons pour faire le projet avec cette logique est d'obtenir des résultats mesurables et utilisables le plus tôt possible.

À la fin de chaque journée, l'équipe de développement de projet organise une réunion rapide d'évaluer ce qui a été atteint, ce qui reste à faire et des facteurs qui peuvent bloquer le projet.

Puisque le résultat d'une timeboxe est souvent un ensemble de maquettes ou de prototypes, des précautions sont prises afin de soigner la planification et de gérer le processus de modélisation et de prototypage efficacement. Les prototypes peuvent avoir des objectifs très différents. Ils seront aussi testés et vérifiés différemment.

Un «prototype métier» développe la compréhension des besoins. D'habitude, un cas ou un scénario d'utilisation, des témoignages et des modèles de processus sont des exemples de prototypes métiers. Un prototype métier peut être mis en œuvre sous forme d'un jeu de rôle ou d'une opportunité d'observation de l'utilisateur, autant que dans un format physique.

Un «prototype d'utilisabilité» explore l'interface entre le client et la solution. Dans le cas d'un produit il s'agit souvent d'une interface physique - écran, tableau de bord, instrumentations - qui permettent à un utilisateur d'interagir avec le produit ou le système. Dans le cas d'un service, cette interface peut être en forme d'un point de vente, d'une vitrine ou d'une instance de communication.

Un «prototype de performance» permet aux besoins « non-fonctionnelles » (autrement connu sous le nom de besoins de « qualité de service » ou « d'ilités», comme la fiabilité, l'accessibilité, la transférabilité, la maintenabilité, la sécurité, ainsi que la vitesse de transaction) d'être étudiés, évalués et testés.

Un «prototype technique» permet d'explorer différentes solutions d'un point de vue technique. Donc, c'est le seul des types de prototypes qui adresse les soucis de l'équipe de développeurs principalement, alors que les autres servent les intérêts des utilisateurs.

Le terme «prototype» est utilisé de façon générique. Dans la pratique, une esquisse peut devenir un croquis qui peut évoluer en dessin, pour devenir dessin, puis modèle, maquette, et enfin prototype, en passant par plusieurs versions éditées avant de devenir le produit définitif.

Dans un projet agile des versions de prototype sont prévues au sein de chaque timebox et les résultats de chaque prototype sont clairement spécifiés autant que les contrôles et les essais qu'y seront nécessaires pour valider et vérifier les prototypes.

| Calendrier d'un plan directeur de projet → | | | |
|--|--|--|--|
| Timebox 1 | Timebox 2 | Timebox 3 | Timebox 4 |
| <i>Prototype A1</i> <i>Prototype B1</i> | <i>Prototype A2</i> <i>Prototype C1</i> | <i>Prototype A3</i> <i>Prototype B2</i> <i>Prototype D</i> | <i>Prototype A4</i> <i>Prototype C2</i> |
| <i>A1 produits</i> <i>B1 produits</i> | <i>A2 produits</i> <i>C1 produits</i> | <i>A3 produits</i> <i>B2 produits</i> <i>D produits</i> | <i>A4 produits</i> <i>C2 produits</i> |
| <i>A1 vérifications</i> <i>B1 vérifications</i> | <i>A2 tests</i> <i>C1 vérifications</i> | <i>A3 tests</i> <i>B2 tests</i> <i>D tests</i> | <i>A4 tests</i> <i>C2 tests</i> |

Un autre perspective sur le management d'un projet agile

4) La gestion de l'avancement

De la même manière que les délais sont fixés pour les projets agiles, les coûts peuvent aussi être limités, de sorte que la progression du projet devient une question de combien de besoins peuvent être livrés dans la limite des contraintes de temps et des coûts. Il est important de prendre notes des réactions en termes de résultats acceptés, d'actions définies, et de besoins identifiés, a précisés ou annulés.

Un projet agile se félicite de nouveaux besoins détectés, à condition qu'ils soient justifiés en termes de valeur ajoutée et en tenant compte d'autres risques. Toutefois, l'équipe du projet doit démontrer qu'elle maîtrise le processus, que le projet crée de la valeur et que l'on est en train de converger vers une conclusion satisfaisant.

Un « burndown chart » est idéal pour afficher les besoins qui ont été définis contre ceux-ci ont été réalisés, dans chaque timebox. Lorsque'il y a plus de besoins produits que de nouveaux besoins identifiés un point de convergence peut être atteint.

Les principaux indicateurs de performance de l'entreprise ainsi que des mesures de la qualité du projet sont indiqués sur le tableau de bord du projet, tels que la satisfaction des utilisateurs et le nombre de modules opérationnels.

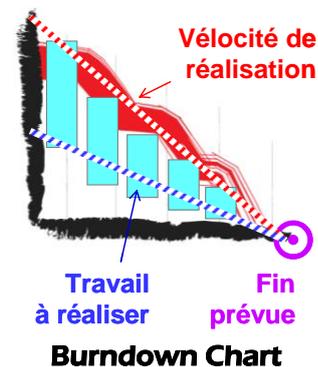
Alors que la solution est en cours d'élaboration, les conventions agiles encouragent les développeurs à partager leur production en tant que copropriétaires des modules en développement. Ce n'est pas obligatoire, mais cela fait partie de la logique de partage et de communication. Tout comme « l'open source » stimule la participation entre pairs et leur coopération, la programmation par les pairs est reconnue comme une source de productivité, car la détection des erreurs est plus évidente et intervient plus en amont.

En outre, une approche agile est facilitée par une bonne conception, une architecture modulaire et la possibilité de réutiliser des composants, des modèles et des prototypes. Des développeurs agiles sont bien conscients qu'une composante bien finie sera beaucoup plus facile à maintenir et à se redéployer dans l'avenir.

La gestion de l'état d'avancement d'un projet itératif et agile repose sur une bonne gestion de la configuration, ce qui impose une maîtrise de la traçabilité et l'intégration des composants.

Un environnement agile est très dynamique. Tout ce qui peut changer ou évoluer doit être géré. Le rôle du coordinateur technique dans un projet agile est l'équivalent du coordonnateur de projet dans un projet traditionnel. Le rôle couvre la gestion de l'impact des changements sur les autres composants, veiller à ce que les modules peuvent interagir ensemble et s'intégrer, et s'assurer que chaque composant est testable et sera testé.

Pendant la durée du projet agile, le processus de développement et de test nécessite des contacts fréquents entre les développeurs et les utilisateurs, facilités par des ateliers. À la fin du projet, les essais d'acceptation sont susceptibles d'être beaucoup plus simple parce que l'entreprise et l'équipe de développement sont en confiance, et les utilisateurs sont plus susceptibles d'approprier les résultats auxquels ils ont participé tout au long du projet. En vérité, l'agilité est un facteur critique de succès pour tout projet technologique.



Un autre perspective sur le management d'un projet agile

5) Comment tester, gérer les risques et clore un projet agile.

À son arrivée chaque matin, l'utilisateur ambassadeur d'un projet agile teste les modules livrés la veille avec des données réelles. Il souligne en rouge les problèmes critiques, en rose les incidents ennuyeux posant des difficultés mais pour lesquels il y a une solution alternative et il laisse en blanc les questions cosmétiques.

Plus tard dans la matinée, après un café, l'équipe de développement examine ensemble l'état quotidien de l'avancement durant une réunion téléphonique de 15 minutes. Elle identifie les obstacles et définit le travail à accomplir pendant la journée.

L'utilisateur ambassadeur ayant dessiné une plaquette projet, il représente la communauté plus large des utilisateurs. Son bureau est agrémenté d'un jardin zen, décoré avec du sable ratisé au râteau, des pierres, et des bonsaïs. Une distraction utile pour les matinées où le travail qui reste à faire semble évoluer plus vite que la réalisation. Des prototypes destinés aux utilisateurs étaient réalisés toutes les deux semaines et présentés autour d'un déjeuner-buffet.

Des mesures ont été fondées sur le nombre de problèmes en suspens ainsi que les demandes par module. Les modules sont considérés comme viables pour l'utilisateur pilote une fois que les points de blocages soulignés en rouge ont été éliminés. Ils sont rendus utilisables par d'autres membres de la communauté des utilisateurs une fois que les difficultés soulignées en roses ont été résolues. Un livret de travail électronique centralise la dernière version du contenu du projet. Les utilisateurs, qui peuvent consulter l'état des demandes restantes et les problèmes, peuvent obtenir un rapport de synthèse sur la maturité de l'ensemble du système ainsi que pour chaque module.

Le contrat est établi sur des modalités forfaitaires, il respecte un délai fixe convenu pour une première livraison à une communauté d'utilisateurs pilotes. Il est fondé sur un ensemble hiérarchisé de besoins. Un avenant est attribué afin de fournir certaines demandes supplémentaires plus vastes à la communauté d'utilisateurs dans une deuxième version.

Après deux années d'utilisation intensive, les utilisateurs demandent une mise à niveau et un transfert vers une nouvelle plate-forme. Dans ce deuxième développement, en raison de l'indisponibilité d'un utilisateur clé, une approche plus classique est adoptée. Les tests sont délégués à un consultant externe, mais le développement prend presque trois fois plus de temps.

Cette expérience révèle quelques-unes des menaces et des opportunités dans la mise en œuvre d'un environnement de projet agile. Sans un niveau adéquat d'implication, de coopération et d'engagement, à la fois des utilisateurs et des développeurs, le projet et le processus peuvent être détournés.

Ce sont quelques-uns des symptômes d'échecs :

- Les principaux participants ne sont pas disponibles pour les décisions ou ne sont pas constructifs pendant les ateliers.
- Lorsque les exigences sont définies ou évoluent, ils ne sont pas associés à des priorités.
- Les développeurs sont incapables d'interpréter les besoins de l'entreprise en termes de produits livrables.
- L'environnement de travail ne permet pas de développement itératif et incrémental.
- Les prototypes et les ateliers ne sont pas justifiables, car les besoins sont déjà bien compris.

Un autre perspective sur le management d'un projet agile

Néanmoins, l'utilisation d'une approche agile permet d'éviter certains des risques traditionnels qui se produisent suite à des malentendus entre les protagonistes du projet:

- Le coût élevé des problèmes, et des échecs parce qu'ils sont découverts trop tard.
- La spécification des fonctionnalités coûteuses, pour se protéger dans un projet long
- Des changements coûteux liés à un environnement et à une architecture qui ne sont pas conçus pour l'agilité
- Des rapports sans visibilité, et sans résultats tangibles vérifiés constamment et validés jusqu'à l'acceptation.
- Un manque d'engagement, de coopération ou de communication conduisant à l'absence d'un sentiment d'appropriation.

Annexe A) Quand utiliser une méthode agile

Il existe une justification de projet qui est solide



Il y aura des besoins qui seront plus importants ou plus urgents que d'autres



Les besoins seront évolutifs et le projet va générer des connaissances



Un dialogue peut être établi entre les réalisateurs et des clients disponibles pour faire des tests

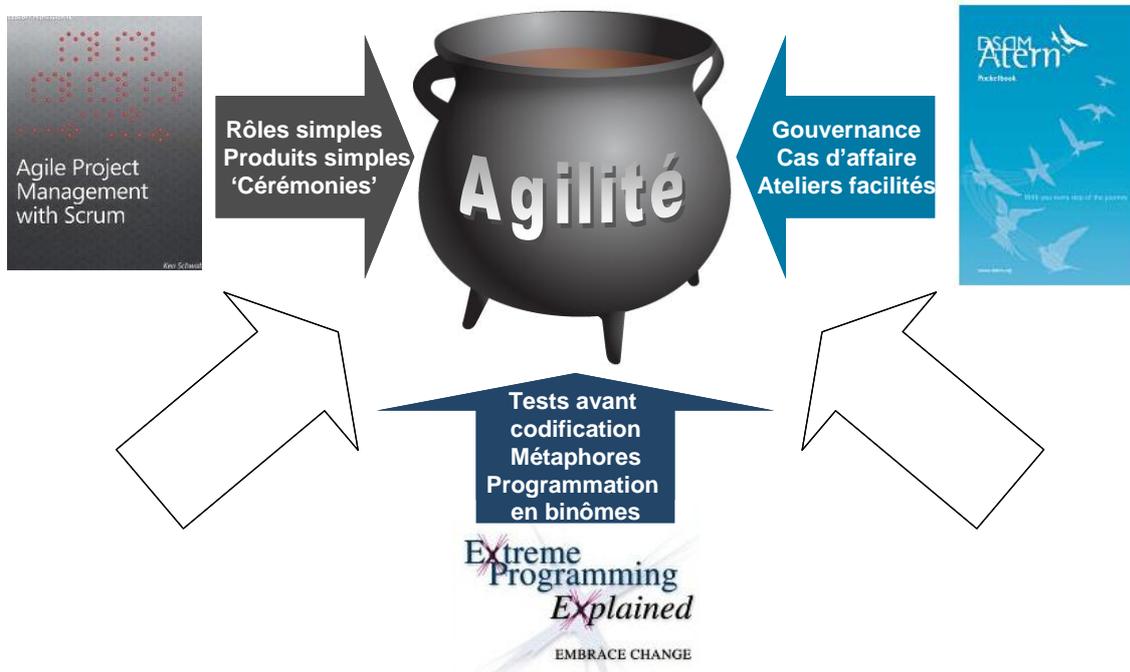


Une architecture et des moyens existent pour pratiquer la modélisation et le prototypage



Un autre perspective sur le management d'un projet agile

Annexe B) Synthèse des méthodes



Annexe C) Une matrice de comparaison subjective des méthodes agiles

