

# PRIMER CONTRATS AGILE

Extrait du livre...

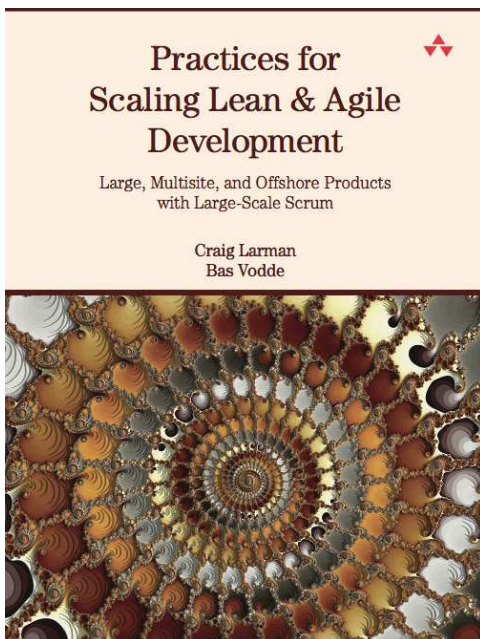
Practices for Scaling Lean & Agile Development:  
Large, Multisite, & Offshore Product Development with Large-Scale Scrum

de Tom Arbogast, Craig Larman et Bas Vodde

Version 5

Merci de nous envoyer vos commentaires pour les versions futures, à [www.agilecontracts.org](http://www.agilecontracts.org)

Note : Vérifier sur le site la dernière version, partager l'URL (plutôt que le fichier) pour rester à jour



1. Introduction
2. Large-Scale Scrum
- Action tools
3. Test
4. Product Management
5. Planning
6. Coordination
7. Requirements
8. Design
9. Legacy Code
10. Continuous Integretion
11. Inspect & Adapt
12. Multisite
13. Offshore
14. Contracts ←

## TABLE DES MATIERES

- INTRODUCTION 3
- PARTIE 1 : REFLEXION SUR LES CONTRATS 4
- PARTIE 2 : SUJETS COMMUNS DES CONTRATS AGILES 18
- PARTIE 3 : MODELE DE CONTRAT 33
- CONCLUSION 40
- LECTURES RECOMMANDEES 41

Note: La structure “Essayer” et “Eviter” de ce chapitre suggère l’expérimentation, pas des formules.

### A PROPOS DES AUTEURS



**Tom Arbogast** est un juriste ayant de nombreuses années d’expérience dans les projets IT et les contrats, combinées avec des connaissances des principes agiles, de la pensée systémique, de la pensée Lean. Il a travaillé des trois côtés de la barrière: (1) comme juriste spécialiste des contrats pour les clients des services informatiques externalisés, (2) comme juriste professionnel des sous-traitants IT («fournisseurs»), et (3) dans le développement des affaires pour les sous-traitants (accords de ventes influençant le contenu du contrat). Tom dirige les équipes des Services Professionnels et Pratique Juridique chez ClearEdge Partners Inc, spécialisées dans l’informatique, Négociation de Fusions et

Acquisitions, Gestion de la production et des opérations. Il a été vice-président de British Telecom et a eu divers postes de direction et de juriste dans les télécommunications et la technologie. Il a travaillé comme juriste pour le privé et soutenu un certain nombre d'affaires devant la Cour suprême du Canada. Il est diplômé de l'Université du Colorado et l'Université de droit de British Columbia.



**Craig Larman et Bas Vodde** sont auteurs des livres : (1) *Scaling Lean & Agile Development: Thinking & Organizational Tools for Large-Scale Scrum* and (2) *Practices for Scaling Lean & Agile Development: Large, Multisite, & Offshore Product Development*

Ils travaillent comme consultants en conception d’organisation et comme coaches en management et ingénierie pour des groupes adoptant la pensée Lean et le développement agile dans le cadre du développement de produits de systèmes de grande envergure, en offshore et en multi-sites. Pour consultation et plus d’information, veuillez

visiter les sites [craiglarman.com](http://craiglarman.com) et [odd-e.com](http://odd-e.com).

[www.agilecontracts.org](http://www.agilecontracts.org)

2/45

Copyright (c) Tom Arbogast, Craig Larman & Bas Vodde 2012 – All rights reserved  
Traduit par Nicolas Mereaux, Laurent Carbonnaux et Pierre Fauvel en décembre 2012

## INTRODUCTION

*L' Histoire me sera indulgente, car j'ai l'intention de l'écrire.*

*-- Winston Churchill*

Les entreprises ont depuis de nombreuses années et avec succès, écrit et utilisé des « contrats agiles » ou « contrats évolutifs ». Chez Valtech (où Craig a travaillé), ils appliquent Scrum sur les projets externalisés dont ils ont la charge, autant dans leur centre de développement de Bangalore que partout ailleurs, et ont écrit des contrats le supportant. D'autres sous-traitants agiles, tels que ThoughtWorks, ont fait de même.

Cette introduction est destinée à deux types d'audiences : les non-juristes et les juristes (du droit contractuel). Nous vous encourageons à la partager avec les professionnels du droit, certains sujets étant écrits spécifiquement à leur intention.<sup>1</sup> – La raison en étant que la majeure partie du travail consistant à créer des contrats supportant les valeurs et pratiques agiles ne réside pas dans le texte du contrat même, mais dans l'*éducation des professionnels du droit* à ces valeurs. Cela implique de comprendre et d'apprécier les problématiques légales traditionnelles, de les prendre en compte, et d'aider les juristes à saisir ce qu'impliquent l'agilité et la pensée systémique. C'est pourquoi les premières recommandations se concentrent sur la *compréhension*. Les sujets suivants se concentrent sur quelques suggestions de cas concrets de contrats-agiles.

### Prudence...

*Pour tout problème complexe, il existe une solution simple, soignée et fausse – H.L. Mencken*

Ne croyez pas que les négociations contractuelles sont bien moins complexes ou vigoureuses pour les professionnels du droit saisissant les implications des principes agiles. Il est important de reconnaître que la contractualisation est un processus compliqué par nature, d'autant plus dans un domaine de haute complexité et d'incertitude tel que le développement logiciel. Et les juristes, par formation et devoir, doivent continuer à prêter une attention particulière aux cadres nécessaires pour faire face à une rupture de confiance et de collaboration entre les parties.

---

<sup>1</sup> Ce chapitre résume les concepts au cœur de l'agilité, familiers des lecteurs experts agiles, tenant compte que les professionnels du droit, nouveaux sur le sujet, sont une audience importante.

## 1ERE PARTIE : REFLEXIONS SUR LES CONTRATS

### Essayer... Partager ces informations clés avec les juristes du contrat

Les points suivants sont essentiels ; ils nécessitent d'être clairement explorés avec les professionnels du droit:

- Les aspects structurels et juridiques des contrats des projets agiles sont les mêmes que pour les contrats de développement plus traditionnel. La principale différence est l'approche et la compréhension des processus opérationnels et de livraison et comment cela est appréhendé dans les contrats ou comment cela impacte les contrats.
- Une compréhension des principes Lean et Agiles et de la pensée systémique est nécessaire pour les juristes du contrat. Pourquoi ? Parce que l'application de ces outils de réflexion conduit à moins de risque et d'exposition, et cela doit être exprimé dans le contrat. Une approche agile permet de fournir rapidement des livrables incrémentaux déployables, des prises de décision collaboratives entre les parties, et donc soulage la pression sur la responsabilité, la garantie, et des questions similaires.
- Les contrats reflètent les espoirs des gens, et spécialement leurs peurs. Les projets ayant réussi ne le sont pas du fait de leur contrat, mais de relations basées sur la collaboration, la transparence, et la confiance. Les contrats à « succès » contiennent les mécanismes favorisant l'élaboration de la collaboration, de la transparence, et de la confiance. Comme la confiance construit la relation entre le client et le fournisseur, le modèle commercial et contractuel doit « s'assouplir » pour favoriser l'accroissement de « la collaboration avec les clients plus que la négociation contractuelle »

### Contrairement à l'intuition...

La priorité numéro une **de tous** est de réussir des projets avec succès (en d'autres termes, réaliser les objectifs de rentabilité), et chaque membre de l'organisation, incluant les professionnels du droit, doit s'efforcer à réduire les optimisations locales, les mentalités en silo, et les gaspillages.<sup>2</sup> D'autres problèmes (de droit) sont importants, mais subordonnés à l'objectif de réussite d'un projet. C'est un changement majeur de pensée parce que de nombreux juristes voient leur *fonction* discrète comme la priorité, qui est, livrer un contrat « à succès ».

### Essayer... Les juristes étudient les concepts agiles, itératifs et de la pensée systémique

Un avocat écrivant un contrat pour un projet agile (le plus souvent, fait en Scrum) doit saisir les idées clés avant de pouvoir articuler un contrat agile. Nous suggérons aux professionnels du droit d'étudier les éléments d'introduction à ces sujets. Par exemple :

- Dans le livre *Agile & Iterative Development*[Larman03], chapitre deux, *Iterative &*

2 Gaspillages: 1. La surproduction de fonctionnalités; 2. Attentes et les retards; 3. Transferts; 4. Réapprendre; 5. Travail partiellement effectué; 6. Commutation de travail; 7. Défauts (et les essais liés, d'inspection et de correction); 8. Personnes sous-utilisées; 9. Perte de connaissance et de dispersion; 10. L'utopie.

*Evolutionary*, et chapitre trois, *Agile*

- *The Scrum Primer* ([www.scrumprimer.com](http://www.scrumprimer.com))
- La section *Continuous Improvement* du *Lean Primer* ([www.leanprimer.com](http://www.leanprimer.com))
- Les articles sur la pensée systémique ; [www.thinking.net](http://www.thinking.net) contenant des articles et de nombreux liens
- Dans le livre *Scaling Lean & Agile Development* (LV08] chapitre deux, *Systems Thinking*
- Cette introduction

### Essayer... Apprécier le point de vue d'un avocat traditionnel

*Les professionnels du droit sont câblés différemment.* Ce recâblage démarre dès qu'ils entrent à la faculté de droit. Les concepts de *Responsabilité Professionnelle* et de *Plaidoyer* sont définitivement ancrés dans la pensée des juristes. Les professionnels du droit sont formés à agir, sous l'obligation **légale**, pour faire avancer les intérêts de leurs clients et les protéger contre tous les pièges, visibles ou invisibles. Comment définissez-vous les intérêts d'un client? *Un client* pourrait probablement dire tout simplement la réussite du projet. *Une professionnel du droit* dirait qu'elle a réussie si elle protège son client au plus haut niveau possible contre l'exposition et le risque, tout en faisant avancer l'objectif final du contrat / projet.

Il n'y a qu'à regarder les définitions légales du devoir d'un avocat pour voir comment un avocat perçoit son rôle:

*(5) Un avocat doit s'efforcer par tous les moyens justes et honorables d'obtenir pour un client le bénéfice de tout recours et moyen de défense autorisés par la loi. L'avocat doit, cependant, résolument garder à l'esprit que cette grande confiance doit être exécutée dans les limites de la loi et non en dehors.*<sup>3</sup>

Les juristes voient dans leur rôle le fait d'être là pour *protéger* leur client de choses qu'ils pourraient ne même pas connaître. Un avocat est ostensiblement éduqué à se *méfier*, pas nécessairement des autres personnes, mais des attentes irréalistes et des résultats (les gaspillages de l'utopie), en particulier au début d'un projet.

Il est important de comprendre cette dynamique dans le contexte d'une négociation contractuelle. Lorsqu'un avocat affirme qu'une partie de son rôle est de traiter (contractuellement) le cas où la confiance se détériore, cela n'implique pas que l'avocat ne fait pas confiance à l'autre partie. Cela veut plutôt dire qu'elle ne fait pas confiance aux *attentes* des résultats escomptés et est mandatée pour faire face à la plupart des résultats attendus, bons ou mauvais.

La troisième valeur du Manifeste Agile est la *collaboration avec les clients plus que la négociation contractuelle*. Naturellement, en première lecture, un avocat va prendre note, réagir et sans doute penser « *C'est bien, mais je suis ici pour assurer que mon client est correctement protégé. Elle peut penser ce qu'elle veut, mais je parie qu'elle ne dirait pas qu'elle valorise la collaboration plutôt que la négociation du contrat quand tout va mal et qu'une action en justice est déposée* ». Il est du

---

3 The Law Society of British Columbia ; Rules of Professional Conduct

devoir de l'avocat d'examiner l' «impensable» dans les relations contractuelles et de fournir un cadre (exprimé dans un langage contractuel) pour traiter les résultats désagréables. Les juristes sont formés, et sur-expérimentés, à traiter ce qui arrive quand les relations se détériorent et quand la confiance se casse.

### Règle du précédent <sup>4</sup>

Les juristes sont des créatures d'habitudes. Cela provient de la façon dont le droit s'est développé.

*La vie du droit n'a pas été la logique, mais l'expérience. - Oliver Wendell Holmes*

Il est dit que la loi a souvent un temps de retard, jamais d'avance. C'est dû à la manière avec laquelle le droit se développe. Les affaires sont portées devant les tribunaux et analysées dans le contexte des principes juridiques en vigueur. Cette idée s'applique à tous les domaines du droit, y compris les principes contractuels acceptés.

Une fois qu'un problème a été examiné et analysé *ad nauseam*, y compris dans les milieux universitaires juridiques, il sera alors accepté dans la pratique courante. Ce processus peut prendre une décennie voire plus.

Les juristes regardent donc les anciens modèles éprouvés, époussetant les vieux précédents et prenant le droit acquis comme un guide. Tout ce qui est perçu comme nouveau ou changement majeur (les méthodes agiles, par exemple) est vu avec scepticisme et méfiance. Et ce dépoussiérage s'applique aux modèles de contrats ; il est plus facile de réutiliser des modèles existants que d'en recréer de nouveaux.

### Hypothèses des projets traditionnels : Impact sur les contrats

Qu'est-ce que les juristes pensent de la nature des projets informatiques ? Premièrement, il est courant qu'ils les voient comme des projets de *construction*, relativement prédictibles, plutôt que des projets de recherche et de développement, hautement incertains et variables. Deuxièmement, parce que lors d'un projet, (1) un long moment est nécessaire avant que quelque chose soit délivrée de manière bien faite, (2) avec de tardifs et faibles retours, (3) de longs cycles de paiement, (4) et de gros problèmes pour le client si le projet s'arrête à n'importe quel moment de manière arbitraire. *Ces hypothèses sont invalides dans les développements agiles.*

Bien sûr, ces hypothèses sont exprimées dans un langage contractuel, et de manière telle que les juristes puissent prendre le temps et l'attention nécessaire pour appréhender les concepts tels que les risques et responsabilité de retard, résiliation, indemnisation, tests d'acceptation, critères de paiement et de garantie, entre autres.

### Essayer... Corriger les malentendus courants lorsque la troisième valeur agile est

---

4 La règle du précédent implique que la règle précédente s'applique et ne seront pas modifiées tant qu'une alternative n'est pas acceptée par les tribunaux, elle s'applique principalement aux pays ayant un système de type « common law ».

NDT : c'est-à-dire le Royaume-Uni (l'Écosse faisant en partie exception), et beaucoup de ses anciennes possessions, notamment les États-Unis et les États du Commonwealth, tels que le Canada (le Québec faisant en partie exception) et l'Australie.

## présentée aux juristes

Comme évoqué précédemment, les professionnels du droit novices aux valeurs agiles vont réagir à la première lecture de *collaboration avec les clients plus que la négociation contractuelle*. Il est utile pour les non juristes de connaître de telles réactions et d'aider à traiter ces malentendus au travers de discussions. Et les juristes peuvent corriger ces malentendus en étudiant :

**Les Fausses dichotomies** – Le premier et sans doute le plus courant des malentendus est l'erreur d'interprétation par une mauvaise dichotomie des valeurs agiles; c'est-à-dire, « la collaboration avec les clients c'est bien, mais la négociation contractuelle c'est mal », plutôt que, citant le manifeste agile, *...nous reconnaissons la valeur des seconds éléments, mais privilégions les premiers*. Les professionnels du droit doivent comprendre qu'au travers de cette valeur, le contrat n'est pas subrogé à l'effort collaboratif, mais plutôt que la collaboration est dominante pour la réussite d'un projet.

La collaboration ne doit pas seulement être exprimée sous forme de comportement entre les parties durant le développement du projet, elle peut et doit être exprimée dans le langage contractuel (et le comportement des juristes). Le contrat peut définir un cadre encourageant les pratiques collaboratives, et en ce sens, les professionnels du droit peuvent supporter les objectifs agiles de leurs clients, et plus important encore, améliorer la réussite des projets.

**Les 'Contrats' non-légaux** – Un autre malentendu est de penser que la troisième valeur vaut uniquement pour les contrats légaux. Mais « la négociation contractuelle » ne fait pas exclusivement référence aux contrats légaux ou des affaires. Elle est destinée à inclure la notion plus large d'accords ou de spécifications entre les parties dans le développement de produits, et savoir si l'accent doit être mis sur *mettre noir sur blanc* ces accords ou sur une collaboration continue, l'apprentissage et l'évolution. Par exemple, une approche traditionnelle inclus très tôt une spécification détaillée des exigences suivis d'une validation, puis ensuite elles sont passées aux équipes de développement pour réalisation (un contrat sur les exigences).

Les professionnels du droit peuvent aggraver ou améliorer, par le langage légal contractuel, une focalisation malsaine sur ces 'contrats' non légaux durant la vie du projet. Par exemple, s'ils ébauchent un contrat contenant des clauses nécessitant la définition et l'engagement sur spécifications des toutes ou la plupart des exigences avant de lancer l'implémentation, le projet ne sera pas agile et l'accent sera porté malheureusement sur la négociation contractuelle (non légale).

### Essayer... Les juristes étudient les problèmes issus de la mentalité en silo et le manque de pensée systémique

La figure 1.1 (de l'International Association for Contract and Commercial Management, IACCM<sup>5</sup>) présente le top dix (sur trente) des termes contractuels clefs auxquels les juristes d'entreprises ont eu affaire entre 2002 et 2007. Il est difficile d'imaginer que les personnes en charge de livrer soient concernées au jour le jour avec la plupart des questions listés. Et il est frappant de voir *qu'une description de l'objet du contrat (l'objectif du contrat) n'est pas mentionnée*<sup>6</sup>. C'est une observation

5 Ndt : [www.iaccm.com](http://www.iaccm.com)

6 Livraison/Acceptation est référencé à l'item-9. Malgré tout, cela référence le concept de livraison, associé à un régime spécifique d'acceptation, et n'est pas concerné par l'objet de la livraison lui-même.

étonnante. La seule véritable chose dont un contrat devrait s'inquiéter (par exemple, un logiciel permettant d'accélérer le traitement des factures) n'est pas dans le top dix.

Figure 1.1 top dix (sur trente) des préoccupations contractuelles des juristes d'entreprise

	Termes du Top 30 2007	▲ ▼	2006	2005	2004	2003	2002
1	Limitation de responsabilité	-	1	1	1	1	1
2	Indemnisation	-	2	2	4	10	3
3	Prix / charge / modifications tarifaire	▲	4	6	3	5	7
4	Propriété intellectuelle	▼	3	3	5	3	2
5	Résiliation (cause / commodité)	-	5	7	7	7	5
6	Garantie	-	6	5	2	2	6
7	Niveau de services	▲	11	10	13	6	6
8	Paieement	▲	9	4	6	4	11
9	Livraison / Acceptation	▼	8	9	8	12	13
10	Information confidentielle / Protection des données	▼	7	8	10	14	15

Considérez ce scénario : Il est demandé à un juriste d'une grande entreprise de « mesurer le succès » des contrats dont le département juridique s'occupe. Le juriste répond, « Nous nous sommes occupés de plus de six milliards de dollars d'obligations au cours des dernières années englobant plus de 400 contrats différents, et nous avons seulement été poursuivi, ou avons poursuivi, sur deux de ces contrats. Ceci est cohérent avec notre suivi annuel de performance, s'élevant à environ un taux de réussite de 99% ».

Dans le monde des juristes traditionnels, c'est leur définition du succès, la situation la « meilleure » ou « optimale ». Mais bien sûr, elle n'est que *localement optimale*. Le juriste ne se préoccupe pas que l'affaire derrière le projet se soit réalisée, que les utilisateurs du nouveau logiciel soient enchantés, que le projet fut livré, ou si trop de choses ont été payées au cours de la vie de n'importe quel contrat en particulier.

Comment les juristes mesurent le succès tout en respectant la négociation contractuelle ? Il est de tradition de dire au sujet des contrats négociés sereinement que, « vous savez que vous avez un bon contrat lorsque les deux parties ne sont pas contentes », parce qu'aucune des parties n'a eu ce qu'elle voulait. La mentalité agile défend le contraire pour les deux parties, et une approche « gagnant-gagnant » est vraiment ce qui est mutuellement optimal.

Mais indépendamment de la façon dont on mesure 'la bonté' d'un contrat, une chose reste constante, et est au centre de la peur qu'un juriste a en ébauchant un contrat : Si quelque chose va mal, le client va regarder le contrat (et donc le juriste) pour s'assurer que la question est couverte en faveur du client. Cette peur, ainsi que les attentes du client, conduit le juriste à optimiser localement



du strict point de vue du client dans le respect des *scenarios de problèmes juridiques*.

Ce qui amène au concept d'optimisation locale, ou la tendance qu'ont les acteurs d'un système complexe à faire de leur « mieux » dans les limites de leurs propres devoirs et rôles, sans comprendre l'impact plus large de leurs choix et actions, ou ignorant les objectifs de plus haut-niveau du système.

La réponse du juriste à la question concernant le succès du contrat était convaincante dans un contexte local mais ne prend pas la mesure des problèmes de plus grands systèmes. Et pourquoi cela ? Parce qu'il y a...

- un large fossé entre les groupes juridiques et ceux de livraison.
- une mentalité endémique de silo parmi les juristes du droit contractuel.
- un manque d'analyse systémique entraînant des optimisations locales.
- des mesures et incitations fondées sur des préoccupations juridiques.

Sur ce dernier point : Mesures et incitations ne font pas qu'injecter des dysfonctionnements et un comportement d'optimisation locale dans la livraison du projet, elles font de même dans l'écriture du contrat. Si les professionnels d'un département juridique sont récompensés sur la base des résultats juridiques, il y aura sans doute moins de problèmes juridiques, mais pas de plus grands succès de projets.

### La forme versus la fonction

Nous avons tous été dans des bâtiments, très agréables et beaux de loin, mais qui sont non-fonctionnels et déroutants à l'intérieure. C'est la différence entre la forme et la fonction. Tout juriste professionnel vous dira que, lorsqu'un contrat est terminé et dépoussiéré, avec l'encre des signatures encore fraîches, le produit final étincelant (qui peut être une pile d'un mètre de documents et d'appendices) est une beauté à contempler.

Mais bien sûr le test réel d'un contrat est dans l'étape d'exécution du projet, lorsque les gens du terrain travaillent ensemble. Durant cette étape, toute nécessité de se référer au contrat est un signe d'échec – non seulement de la collaboration mais aussi dans la capacité des juristes professionnels à favoriser un cadre de collaboration et de succès.

Ce qui veut dire, que si une référence au contrat est nécessaire, c'est là où il acquiert une vie propre, un peu comme un immeuble. *Il est donc essentiel d'envisager à quoi va ressembler le contrat dans l'usage quotidien.* Cette vision va de pair avec une approche de pensée systémique.

Tout ceci entraîne donc la question : Combien de temps est dépensé pour la négociation des différents domaines du contrat ? Est-ce que les juristes professionnels optimisent localement les préoccupations et le langage du contrat (reflétant leur mentalité de silo) sur de nécessaires mais secondaires questions, et en conséquence augmentent réellement le risque d'échec des projets ?

Beaucoup de juristes passent une quantité de temps excessive et rencontrent des problèmes sur des aspects légaux du contrat (par exemple, dépensant des heures incalculables sur les domaines tels que la force majeure et la responsabilité). Il est certainement important de considérer ces aspects, mais quelle est leur importance dans une vision plus large visant à assurer le succès de l'élément

sous jacent du contrat : le projet ?

Il y a une histoire incroyable [Parkinson57] racontée par le fonctionnaire britannique, C. Northcote Parkinson, illustrant sa *Loi de la Banalité* : Le temps passé sur tout sujet d'un ordre du jour est inversement proportionnel au coût du sujet. Il raconte l'histoire d'un comité de pilotage gouvernemental avec deux sujets à l'ordre du jour : 1) le choix de la technologie pour une centrale nucléaire, et 2) le choix du café pour les réunions. Les mandarins du gouvernement, accablés par la complexité technique et la science, sont passés rapidement sur la recommandation technologique de l'ingénieur conseil, mais tout le monde a eu une opinion sur le café et a voulu en discuter longuement.

Une dynamique similaire se joue entre les juristes écrivant des contrats pour des projets : Il y a une relation inversée entre le temps passé sur les termes en cours de négociation et ce à quoi l'on doit faire face au jour le jour pendant l'exécution du projet.

Mais il y a de bonnes nouvelles respectant les enjeux de la négociation : Une approche Agile et itérative peut, par conception, diminuer le risque. Par conséquent, la pression sur la négociation des termes de "questions critiques" (comme la responsabilité) est atténuée car les méthodes agiles impliquent des livraisons précoces et fréquentes de parties *finies* du système. Les premiers retours et la livraison d'un système fonctionnel toutes les deux semaines (par exemple) change fondamentalement la dynamique qu'il y a derrière la négociation de certains termes, dont les interminables négociations dans les projets traditionnels «en cascade» sont entraînées par l'hypothèse (et la crainte) d'un long délai avant la livraison.

On peut comprendre comment une pression extrême s'exerce sur les termes d'un contrat, quand ils sont vus sous l'angle d'un modèle de livraison « tout ou rien ». Dans le cas de livrables de petite taille et de nature itérative, dans une approche agile, avec la possibilité d'arrêter le projet à n'importe quelle période de 2 semaines (puisque chaque tranche incrémentale du système est finie et potentiellement déployable), il devrait y avoir moins de pression sur des concepts comme les responsabilités multiples et l'indemnisation.

Dans « The Fifth Discipline », Peter Senger indique que *la pensée systémique et une organisation capable d'apprentissage ont pour but ultime de construire « ... des organisations où les gens vont continuellement accroître leur capacité à obtenir les résultats qu'ils désirent vraiment, où les motifs de pensée nouveaux ou expansifs sont favorisés, où les aspirations collectives sont libérées, et où les personnes apprennent continuellement à apprendre ensemble »*. Dans ce contexte, il est critique pour les professionnels du droit d'admettre que le contrat du projet, même s'il est nécessaire, est secondaire en regard de l'extension de cette capacité. Il est donc critique pour le département juridique d'admettre que les contrats qu'il produit peuvent trop souvent dégrader le succès des projets et dégrader l'apprentissage organisationnel à cause d'un manque de pensée systémique, d'une mentalité en silo et d'une optimisation locale sur des considérations secondaires— il en est d'ailleurs de même pour les autres départements tels que les départements financiers et les ressources humaines.

## Essayez ... les avocats étudient l'impact d'incrément potentiellement déployables toutes les deux semaines

### La Lexus LS contre la Lexus IS



Les projets traditionnels non-agiles envisagent un produit fini semblable à l'acquisition de la voiture très haut de gamme Lexus LS. La solution finale livrée possède toutes les caractéristiques de luxe, polie avec soin. Et – en cohérence avec la métaphore de la voiture – les avocats ont probablement envisagé une approche d'implémentation dans laquelle l'on construit le châssis, puis le moteur est mis en place, suivi de la carrosserie et de l'électronique, enfin l'intérieur et la peinture. Ainsi vous ne voyez le produit fini assemblé uniquement à la fin, lorsque tous les composants sont assemblés.

La pression correspondante mise sur les mécanismes contractuels conçus pour protéger de l'exposition est énorme. Pour un client, cela signifie qu'il y a un régime d'acceptation utilisateur, long et complexe qui doit se dérouler après la livraison *finale*. Et cela signifie que le client ne peut s'assurer de la qualité de la voiture avant qu'elle ne soit *définitivement* livrée. Dans les projets informatiques, cela signifie que le client voudra avoir le maximum de protection pour le périmètre global du projet – habituellement une responsabilité multiple sur le coût global du projet. Ceci signifie que le fournisseur n'est pas complètement en situation confortable avec le livrable jusqu'à la fin du projet, et de cette manière, ne peut être capable de reconnaître la valeur totale de la commande avant la livraison finale.

Un projet agile répond à ces deux problématiques. Il vise non pas à construire itérativement des composants partiels d'un projet, mais plutôt à construire un modèle opérationnel déployable, ayant de la valeur pour le client et qui peut être accepté et utilisé à chaque itération toutes les deux semaines. Ceci est un point critique pour les professionnels du droit novices dans des concepts agiles qu'ils ne maîtrisent pas forcément ; ils peuvent mal interpréter le développement agile comme la livraison incrémentale de composants *non déployables* plutôt qu'un modèle agile de livraison d'un système *utile déployable* après chaque courte itération apportant graduellement de plus en plus de fonctionnalités.

Après la première itération, le modèle ou la solution déployable peut être caractérisé comme une Lexus IS – un simple véhicule d'entrée de gamme. A chaque livraison itérative, le niveau du modèle opérationnel s'élève en fonctionnalités et stature. En un sens, c'est comme la reprise de l'ancien modèle toutes les deux semaines<sup>6</sup>.



L'implication de cette approche est critique pour des concepts comme la responsabilité et l'exposition aux risques. Le client possède quelque chose de valeur pour lequel il a payé et qu'il a acceptée. Le fournisseur peut être confiant : il a livré quelque chose de valeur dont on peut reconnaître la valeur et le prix. S'il y avait, *le ciel nous en préserve*, une rupture dans la relation et que le projet aille *en enfer*, chaque partie sera presque entière en terme d'exposition<sup>7</sup>. Le

6 Cette analogie est imparfaite : Au contraire des voitures, les logiciels – et les contrats – peuvent évoluer vers quelque chose de plus grand à chaque cycle de perfectionnement.

7 Cette analogie simplifiée n'aborde pas les problèmes des coûts attendus, les dégâts conséquents, les manques à gagner, et les autres dommages.

client ne sera pas abandonné en ayant payé pour un projet partiel qui n'est rien de plus qu'une coquille vide, et le fournisseur ne sera pas laissé avec un travail effectué pour lequel il ne sera pas payé. Certes, les ultimes attentes de chaque partie peuvent ne pas être atteintes, et le système partiel peut ne pas avoir assez de fonctionnalités utiles pour être déployé utilement<sup>8</sup>, mais d'un point de vue purement affaires et d'exposition aux risques, les intérêts relatifs des parties ne sont pas autant atteints qu'ils peuvent l'être dans scénario d'un projet traditionnel « en cascade ». *Il est vital pour les avocats d'affaires d'apprécier les implications de ce point quand ils perçoivent, négocient et élaborent les contrats de projets !*

### **Essayer ... les avocats étudient comment l'agilité réduit les risques et l'exposition**

Il y a trois sujets généraux concernés à aborder lors de l'élaboration d'un contrat<sup>9</sup> :

- risque et exposition (responsabilité)
- flexibilité afin de permettre le changement
- clarté sur les obligations, les livrables et les attentes

Un contrat de projet agile posera les mêmes limitations sur la responsabilité (et les termes associés) qu'un contrat de projet traditionnel, néanmoins le contrat agile offrira plus de moyens pour éviter les problèmes mêmes pour lesquels un avocat se fait du souci. Ainsi, une approche contractuelle adaptée aux méthodes agiles *réduira* en fait les risques et *mettra en avant* les intérêts du client. Un contrat qui n'aborde pas les méthodes agiles peut vraiment avoir l'effet inverse à ce qui est attendu, et *accroître* le risque et *taire* les intérêts du client.

Un exemple de cette problématique, est la partie du recensement des besoins et la réalisation des tests du logiciel qui valideront ces besoins. Dans un projet de développement séquentiel, l'avocat renforcera (à travers la rédaction du contrat) les souhaits du client en exprimant chaque cas possible et les tests les accompagnant pour répondre aux besoins qui ont été anticipés.

Une approche agile considère que les exigences seront exprimés d'une manière itérative et évolutionniste afin que le temps et l'argent ne soit pas gaspillé en développant un logiciel dont les besoins ne sont pas finalement nécessaires. Il est aussi reconnu que l'argent pourrait être mieux dépensé pour des exigences non connues au début du projet. Les besoins identifiés et développés dans un projet de développement séquentiel ne seront peut être jamais utilisés, parce qu'ils étaient mal conçus ou qu'ils n'étaient pas le fruit d'une véritable collaboration avec de vrais utilisateurs. Et après la livraison d'un système « conforme au contrat », des exigences ont toujours besoin d'être ajoutées pour répondre aux vrais besoins. D'un point de vue contractuel, cela signifie qu'un contrat basé sur une approche séquentielle augmentera réellement le risque qu'un client paye plus et obtienne moins que ce qu'il attend, et que l'inverse se produit lorsque l'approche agile est comprise et abordée dans un contrat.

Ce point ne peut être surestimé, à la fois d'un point de vue juridique et financier. Dans un projet de

---

8 Un système partiel de bonne qualité rend les choses plus aisées à reprendre et à continuer, pour un autre groupe de développement.

9 Clairement, il existe de nombreux aspects d'un contrat, mais ils peuvent en général être classés dans ces trois catégories.

développement séquentiel, un client pourrait payer plus que le coût anticipé pour obtenir ce qui était initialement attendu. Le contrat en service ne protégera pas contre ce scénario mais en réalité, le promouvra en présumant incorrectement qu'il est tout à fait possible de définir et délivrer un large ensemble de besoins sans des retours continus et sans une évolution de la compréhension du besoin.

Les contrats promouvant ou obligeant un cycle de vie de développement séquentiel accroissent le risque du projet.

Pour un praticant du droit, les implications sont que les principes agiles peuvent protéger le client de choses qu'ils peuvent ne pas connaître. Ces implications sont attachées fermement avec l'affirmation vue précédemment du devoir perçu de l'avocat à son client. D'où, une fois qu'un avocat connaît les principes de l'agilité, il sera négligent s'il ne protège *pas* les intérêts de son client en continuant de permettre (en continuant d'écrire des contrats traditionnels) que son client paye pour quelque chose dont il n'a pas besoin et qu'il paye encore pour comprendre ce dont il a vraiment besoin.

Cela signifie qu'une approche agile

- réduit le risque car il limite à la fois le périmètre du livrable et l'étendu du paiement
- permet les changements inévitables
- se focalise sur les négociations sur les éléments, négligées, de la livraison

Un impératif initial est d'avoir des gens du métier (pour le nouveau système) prêt à mettre la main à la pâte et des membres de l'équipe de développement impliqués et collaborant étroitement tout au long de la vie du projet. Les professionnels du droit sont encouragés à chercher les signes que les parties ont cette intention à l'esprit et à l'encourager durant la phase de négociation et de rédaction.

Cette collaboration continue du client et du fournisseur ne signifie pas que l'avocat ait de grandes opportunités pour d'autres facturations (ou si l'une est interne, c'est beaucoup plus de travail que nécessaire) grâce à l'accroissement des interactions et des découvertes conjointes. Cela signifie plutôt que les modèles contractuels doivent être créés pour permettre une participation, une évaluation et évolution continue du client. Si le bon modèle est créé, un avocat peut avoir un minimum d'implication future – au moins, jusqu'à ce qu'un conflit ne surgisse – car le bon cadre sera en place et facilitera la coopération inhérente dans une approche agile.

### **Essayer ... Élever la sensibilité de l'avocat à la complexité du développement logiciel par analogie avec les affaires juridiques**

Si vous êtes un coach ou un responsable intéressé à éveiller l'intérêt des professionnels du droit à l'aspect variable, naturellement complexe et de découverte permanente des projets informatiques, essayez de partager avec eux cette expérience vécue :

« Je veux un projet de contrat complet pour mon nouveau projet : un nouveau système de gestion financière d'entreprise impliquant 200 développeurs dans 6 pays de 4 prestataires de services avec lesquels nous n'avons jamais travaillé et dont la durée de réalisation devrait être entre 2 et 4 ans. A l'heure près, combien de temps cela va-t-il vous prendre pour négocier et rédiger le contrat avec ces 4 prestataires ? Au mot près, combien de mots contiendra ce contrat ? Quel en sera le coût exact ? »

Discuter du parallèle entre ce scénario et le développement logiciel, et comparer ce qui est réaliste et irréaliste, et les façons efficaces et inefficaces de gérer l'inattendu, la découverte et la variabilité.

Un avocat dira certainement que dans ce cas il est impossible de dire de façon certaine, quel que soit le degré de certitude, à quoi ressemblera le contrat à la fin, à cause de la nature évolutive de la négociation et de l'élaboration du contrat. L'avocat peut être capable de donner un chiffre approximatif arrondi du temps que cela prendra, généralement, « le temps de négocier un contrat complet, prendra, disons, 200 heures », mais il ne s'engagera sur rien de concret en terme de nombre d'heures réelles. Pourtant, ironiquement, les avocats et les responsables métiers (dans un esprit de méthode en cascade) s'attendent à ce que les personnes de l'informatique soient capables d'établir, à travers l'analyse des exigences et l'articulation, à quoi un projet informatique – quelque chose d'une bien plus grande complexité, taille et variabilité qu'un « projet de contrat » – ressemblera et combien il coûtera, avec bien sûr un haut degré de certitude.

### Eviter ... Incitations et Pénalités

Il est habituel pour ceux impliqués dans les contrats (les professionnels du droit, les commerciaux, les acheteurs, ...) de dépenser un temps considérable à inventer, négocier, et écrire des clauses incitatives et des clauses pénalisantes dans les contrats. Il existe une croyance et une présomption incontestée, que les incitations (liés à la performance ou aux dates de livraison) et les bonus sont bénéfiques. Néanmoins, cela est incohérent avec les recherches basées sur des faits réels sur le management [PS06, Austin96, Kohn93, Herzberg87], et il existe de nombreuses preuves indiquant que les incitations conduisent à un accroissement de la course à la récompense, à une réduction de la transparence et de la qualité, et à d'autres dysfonctionnements. Ces recherches sont résumées dans le chapitre *Organisation* de notre livre *Scaling Lean & Agile Development*.

Les pénalités (ou incitations négatives) conduisent aux mêmes problèmes.<sup>10</sup>

Les incitations et les pénalités favorisent la compétition eux-nous entre les clients et les fournisseurs, au détriment de la coopération.

Des alternatives ?

- La simplicité – aucune incitation ou pénalité en fonction de la performance
- Si le client est extrêmement insatisfait avec la performance, arrêter l'engagement à la fin de l'itération
- Les modèles de pertes/profits partagés

### Essayez ... Pertes/Profits partagés

Des modèles partageants la perte ou le profit sont présentés dans les sections suivantes, telle que la section « Essayez ... les contrats à coûts cibles » p36. Cela peut favoriser la collaboration et l'amélioration des deux parties. Par exemple, dans un modèle à coût cible, si le coût réel est plus bas

---

<sup>10</sup> Nous ne faisons pas référence aux pénalités majeures pour négligences flagrantes, mais aux pénalités liées à la variation de la performance

que la cible, le client paye moins et la marge de profit du fournisseur est plus élevée.

### **Evitez ... « le plan de gestion de la qualité » et « la liste des livrables »**

Les travaux traditionnels contractuels externalisés impliquent un faible niveau de transparence et de confiance, et un long délai avant que le logiciel soit fini. Une (des nombreuses) réponses classiques contractuelles est de rendre obligatoire un « plan de gestion de la qualité » conventionnel ou une « liste des livrables » définissant une longue liste de documents à fournir, plutôt que de se concentrer sur la livraison de quelque chose ayant vraiment de la valeur : un logiciel opérationnel. Un négociateur a partagé avec nous, l'histoire suivante :

*Un agiliste impliqué dans la négociation d'un contrat se doit d'être sceptique à propos des documents supplémentaires demandés et doit argumenter pour mesurer le coût de cette production – mais il doit bien sûr être prêt à discuter sur le pourquoi est-ce qu'une organisation en a besoin. Mon expérience (avec la liste des livrables) varie, dont le pire cas fut une très longue négociation avec une entreprise ayant un énorme schisme interne entre les gens du métier pro-agile et les internes traditionalistes de la TIC. Un os à ronger avait été jeté au groupe interne TIC par les gens du métier, qui étaient les négociateurs en chef avec nous, en donnant aux gens de la TIC leur accord qu'un « plan de gestion de la qualité » serait négocié. Le représentant de la TIC essaya de ré-instituer une démarche en cascade et des documents, et un contrôle-et-commande traditionnel, à travers divers brouillons du plan « qualité ». Heureusement, nous avons finalement réussi à éliminer à la fois le plan qualité et le rôle autoritaire et improductif sans-valeur du « responsable qualité » que le représentant de la TIC essayait de bâtir pour lui-même.*

Qu'est ce qui pousse le besoin (assumé) d'une liste de livrables, si tant est que ce soit nécessaire ? Dans Scrum, il existe la « Définition du fini » que les équipes du fournisseur et le product owner du côté du client – plutôt que le responsable de la TIC ou un professionnel du droit – définissent et font évoluer à chaque itération. Les avocats du droit contractuel ont besoin de comprendre la définition du fini parce que cela change la manière dont les contrats agiles sont bâtis, et comment les projets sont réalisés. Pour faire court, la définition du fini de Scrum définit le niveau de finition d'un incrément du produit à chaque itération, ceci en terme d'activités, et d'artefacts, et ceci de telle manière que le produit soit potentiellement utilisable ou livrable à chaque itération. Par exemple, la définition du fini d'un produit particulier d'une itération spécifique incluait « codé, intégré, test fonctionnel/de performance/d'usabilité, documenté » (une vraie définition est plus longue, avec plus de détails et moins d'ambiguïté). Pour réitérer, le « fini » est redéfini par le fournisseur et le client au commencement de chaque itération, et évoluera de manière adaptative avec ce que les parties auront appris comme étant vraiment important.

### **Essayez ... Collaborer au plus tôt et souvent avec les avocats**

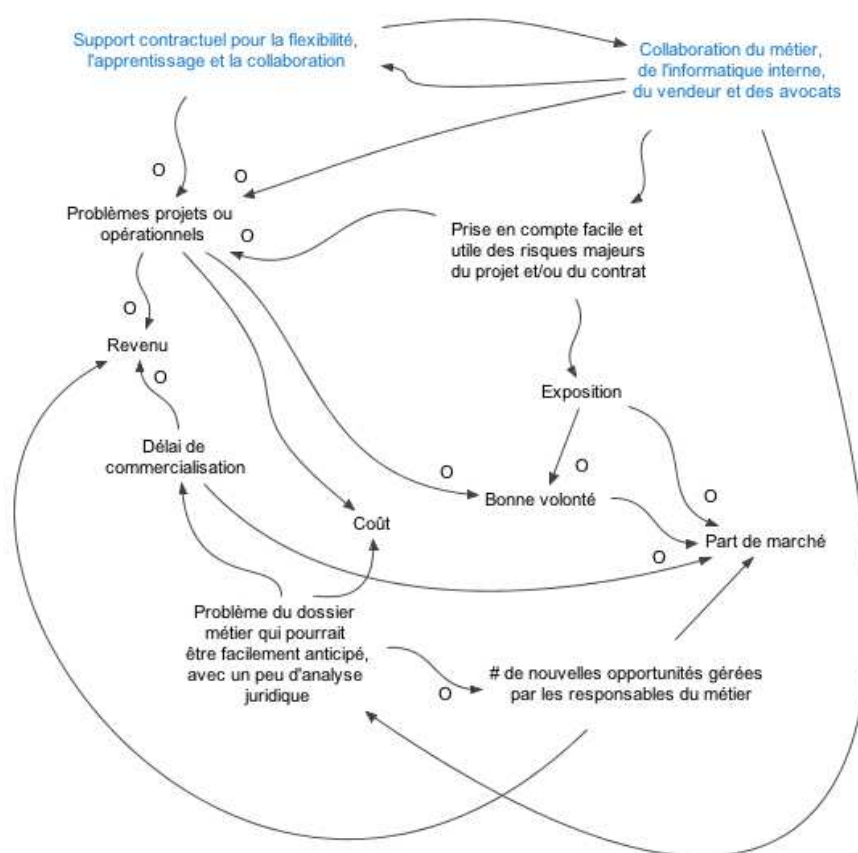
*La collaboration plutôt que la négociation et plus de retours d'informations et plus tôt s'applique non seulement au client et au fournisseur dans un contrat de projet de développement agile externalisé – ils s'appliquent aussi à l'engagement avec les professionnels du droit.*

Le modèle des dynamiques du système de la figure 1.2 illustre, dans les grandes largeurs, les conséquences possibles d'un plus grand support de la flexibilité et la collaboration dans les contrats.

Mais ce qui est surtout pertinent dans cette section, est l'illustration par la figure 1.2 de l'impact de plus de collaboration et plus tôt avec les avocats dans les entreprises et les projets. Cet engagement plus fort fait partie d'un plus grand thème exploré dans le chapitre *Equipe* du livre *Scaling Lean & Agile Development : cross-functional teams*. Les gens présument souvent – à tort - que la frontière d'une équipe pluridisciplinaire sont les personnes appartenant au département R&D ou TIC. Pas vraiment. Par exemple : *Avoir une équipe pluridisciplinaire, signifie que les membres de l'équipe possèdent toutes les fonctions clés généralement présentes dans un projet : Ingénierie, Marketing, et Fabrication, à minima [Smith07]*

En plus de ce minimum requis, l'inclusion du Juridique est nécessaire.

Figure 1.2 dynamiques du système du degré de flexibilité et d'une collaboration au plus tôt avec les avocats.



Ce qui suit, est un cas où j'ai (Tom à la rédaction) vu que l'impact de l'implication tardive de l'avocat – et l'esprit de silo – fût coûteux : les responsables métier identifièrent des pistes de marketing et de baisses des coûts en créant un nouveau système de facturation sur le web. Le dossier d'affaires reposait sur un développement bon marché – qu'ils croyaient pouvoir réaliser par du développement externalisé. En fin de compte, après un processus d'appel d'offres, le finaliste fût choisi et les parties



commencèrent les négociations. C'était et c'est généralement l'étape pendant laquelle les professionnels du droit sont impliqués dans l'élaboration des termes et conditions du contrat.

A l'insu des responsables métier, beaucoup de juridictions ont, récemment, durci les règles sur l'export des données personnelles au-delà des frontières nationales. Cela fût seulement apparent que lors de l'implication malheureusement tardive des avocats. La délocalisation fût infaisable à cause de cette réglementation. Le cas d'affaires – et le projet – furent invalidés.

Heureusement, l'erreur fût rattrapée avant qu'il ne soit trop tard. Mais cela aurait été facile – étant donné l'implication limitée et isolée des avocats – de passer complètement à côté de ce problème, ce qui aurait conduit (comme illustré à la figure 1.2) à une exposition accrue de l'entreprise. Et même si cela a été rattrapé avant le commencement du projet, le travail initial a consommé des ressources significatives de l'entreprise. En plus du gâchis du travail abandonné, les faits consécutifs à devoir reprendre le travail déjà accompli, d'un nouveau dossier d'affaires et d'un nouveau cycle d'appel d'offres ont réduit le temps et l'énergie que les responsables métier avaient pour explorer de nouvelles opportunités d'affaires.

En plus de la valeur des équipes pluridisciplinaires incluant des professionnels du droit, ce cas illustre un point que les personnes de la TIC n'ont pas forcément en tête : un avocat des affaires a le devoir de considérer *deux* types de risques :

- des risques internes au projet
  - ils sont réduits avec le développement agile, et il appartient aux professionnels du droit de supporter cela contractuellement
- les risques provenant d'éléments pouvant avoir des répercussions (tels que la violation de l'export des données)
  - ils sont réduits par des équipes pluridisciplinaires incluant des professionnels du droit, sur la base d'une collaboration régulière et au plus tôt.

## 2EME PARTIE : ELEMENTS COMMUNS DES CONTRATS AGILES

### Pourquoi n'y a t'il pas d'exemples de contrats agiles ?

Lors de l'élaboration de cette introduction, nous avons d'abord considéré inclure des exemples de clauses des contrats agiles créés à Valtech, ThoughtWorks, et d'autres entreprises. Il existe de nombreux exemples de modèles professionnels, en plus des variantes des modèles de contrats de DSDM ou du norvégien PS-2000.

Toutefois, le retour des avocats ayant revus les brouillons de cette introduction, et l'opinion de Tom, notre co-auteur, étaient sans équivoque : le *copier-coller* est un danger présent et réel parmi les avocats et les commerciaux, qui – au lieu d'appréhender les principes du domaine spécifique sous-jacents (tel que les principes du lean ou de l'agilité) inclus dans les éléments contractuels – copieraient-colleraient ces clauses pour élaborer de nouveaux contrats. Les professionnels du droit impliqués dans cette introduction ont un message clair : se concentrer sur les principes permettant l'apprentissage à la fois par les professionnels de l'IT et ceux du droit, qui sont à l'intersection du droit des contrats et du développement agile et lean. Des exemples de clauses cacheraient ce qui est important et inviteraient à ne pas chercher la compréhension profonde et la pensée systémique que les avocats du droit contractuel auront besoin de développer.

### Éléments abordés

Les *éléments* majeurs des contrats de projets agiles (telles que l'acceptation et la fin) sont les mêmes que pour les contrats de projets traditionnels. Toutefois, le contenu de ces sujets dans le contrat – et la volonté des professionnels – incluent des éléments soutenant la collaboration, l'apprentissage, et l'évolution<sup>11</sup>.

L'agilité implique « répondre au changement plutôt que de suivre un plan » et « une collaboration avec le client plutôt qu'une négociation contractuelle », comment cela impacte t'il les éléments suivants du contrat ?

- |                           |                          |
|---------------------------|--------------------------|
| – cycle de livraison      | périmètre du projet      |
| – contrôle du changement  | fin                      |
| – acceptation             | livrables                |
| – périodicité du paiement | tarification             |
| – garantie                | limite de responsabilité |

### Cycle de livraison

Pour les professionnels du droit novices en développement agile, il est impératif de comprendre le

<sup>11</sup> Le cas spécifique des contrats à prix fixe, périmètre fixe sera abordé plus tard

nouveau cycle de livraison. Le cycle, depuis le début, est simplement ceci :

- A la fin de chaque itération d'une durée fixe de 2 semaines (ou jusqu'à 4 semaines), est livré un système déployable avec des fonctionnalités utiles opérationnelles.
- Il peut inclure des fonctionnalités insuffisantes pour présenter un intérêt à être déployé, mais cet *intérêt croît* à chaque cycle.

La livraison incrémentale n'est pas un nouveau concept dans les contrats. Beaucoup identifient les jalons intermédiaires, qu'ils soient fixés par date ou par objectifs avec les acceptations associées ou l'état d'avancement des travaux. Pour les professionnels du droit, les différences notables à appréhender concernant le cycle de livraison et les jalons dans le développement agile incluent :

- le degré de finition et la déployabilité – chaque itération est *finie*, programmée, testée, et ainsi est en théorie déployable.
- la durée – plus réduite, généralement deux semaines
- une boîte de temps – fixée dans le temps mais à périmètre variable

### Périmètre de projet

Les contrats agiles ne définissent pas un périmètre de projet exact et figé, bien qu'il puisse y avoir des variations dans les spécificités du périmètre et des changements, allant de faible à élevé. Comme nous le verrons, ces variations sont, généralement, liées au plan de tarification.

D'un côté du spectre, nous avons les contrats à coûts cibles, dans lesquels le périmètre global du projet et les détails sont identifiés au démarrage, du mieux que l'on peut (afin d'établir le coût cible de départ), mais avec des mécanismes de changement tout au long du projet. De l'autre côté sont les contrats progressifs, dans lesquels aucun périmètre (nécessaire) n'est défini au-delà d'une itération.

### Résumé de la vision – dans le contrat

Il y a des exemples de contrats, dans lesquels le périmètre, la vision et la motivation économique du projet ou du produit est tout à fait impénétrable. Cela doit être évité car cela suggère que les rédacteurs du contrat ne sont pas impliqués dans le projet. Cela relève plutôt d'une mentalité formaliste isolée, une optimisation locale poussée à l'extrême – un défaut discuté précédemment. De plus, un contrat sans une vision projet est moins compréhensible.

Aussi, invitez les professionnels du droit à écrire de manière créative, à partir de leur propre compréhension – pas un copier-coller – un état de la vision selon le style de Moore [Moore91]. Pour achever ceci, ils auront besoin de participer à la vision du projet (par exemple, durant un atelier) et d'autres activités liées au projet. Incluez aussi, un résumé général du contrat, du prix, et du modèle de paiement. Mettez les deux dans le préambule du contrat. Par exemple :

***Pour la comptabilité et le marketing – Qui veut consolider les factures, réduire les coûts de facturation, et cibler le marketing avec des factures – Notre nouveau produit TueurDeFactures est un nouveau système de facturation – Il offre une présentation et un paiement des factures en ligne, ainsi qu'une offre commerciale ciblée. A la différence de notre système actuel de facturation – Notre nouveau produit est sur le web et a un coût de***

*fonctionnement inférieur de 80%.*

*Modèle contractuel: Ceci est un modèle à coût cible. La base est un prix attendu de livraison de \$YYY. Le fournisseur délivrera et sera payé pour, sur la base d'un incrément toutes les itérations de deux semaines.*

## Gestion du changement

La philosophie globale de l'approche agile adresse de façon inhérente la problématique du changement au travers du backlog repriorisable et du planning itératif adaptif. Aucun processus (traditionnel) de gestion du changement, de comité, de mécanisme de demande n'est nécessaire. En effet, il est essentiel pour les professionnels du droit d'expurger des contrats le vieux langage de la gestion du changement car ce langage peut violer l'essence de l'agilité.

Cela ne signifie pas que toutes les formes de gestion du changement sont bannies au sein d'un document contractuel. Certains concepts pertinents dans les contrats de projets agiles sont présents dans deux catégories :

- changement de relation entre les parties
  - Par exemple, lorsqu'une partie est rachetée par une autre entité, un changement fondamental dans la direction de l'entreprise peut se produire. Alors, le langage de gestion du changement couramment utilisé dans les contrats est toujours applicable. Néanmoins, garder à l'esprit qu'à cause de la nature itérative des livrables et du paiement concurrent inhérent à l'approche agile, il y aura moins de pression sur les attentes relatives et sur l'impact qu'aura sur elles un changement majeur.
- changement dans le périmètre du projet
  - Cette partie requiert le plus de soin dans le contrat, pour éviter de corrompre le développement agile : faire des changements rapides et faciles dans un esprit de collaboration entre le client et le vendeur. Eviter les comités nommés pour la gestion du changement, les demandes de changements, ou les processus spéciaux de changements.
  - Mais, comme avec le périmètre du projet, il y a des variations dans la flexibilité de la gestion du changement, allant d'une grande flexibilité sans pénalité lors de l'utilisation de contrats progressif à périmètre flexible, à une flexibilité moyenne avec les pertes/profits partagés lors de l'utilisation de modèles à coûts cibles.

Consulter également « Achèvement »

## Achèvement

Le concept d'achèvement est lié au contrôle du changement, dans ce qu'un projet agile devrait être capable de changer sa course, au point de pouvoir stopper tout effort supplémentaire à la fin de n'importe quelle itération. En contraste à une pensée de projet conventionnel, les professionnels du droit ont besoin de comprendre qu'une fin précoce devrait être perçue comme un événement positif, désirable d'un projet agile, parce qu'une fin précoce ne signifie pas forcément un échec – cela peut signifier que le succès a été rapidement atteint.

Si possible, le modèle idéal d'achèvement dans un contrat agile est de permettre au client d'arrêter, sans pénalité, à la fin de n'importe quelle itération.

Naturellement, si le vendeur a dédié 100 personnes en prévision pour deux ans, et que l'achèvement se produit inopinément plus tôt, il aura probablement un problème coûteux sur les bras. Aussi, les variations sur les clauses d'achèvement de projets agiles incluent une échelle de pénalité pour le client se réduisant au cours du temps (et des itérations).

[www.agilecontracts.org](http://www.agilecontracts.org)

21/45

L'achèvement est sans doute un des aspects les plus difficiles à négocier dans n'importe quel contrat. La clé pour atténuer les différences dans une approche agile est que (1) le client a un système opérationnel à chaque itération, et (2) chaque partie aura une vision claire et à jour sur l'état du livrable. Ce sont des points cruciaux à appréhender pour les professionnels du droit.

### Acceptation

« Est-ce fini ? » – « Que faire si ce n'est pas fini, » – « Nous avons décidé maintenant de changer d'avis et de rejeter la livraison des trois itérations précédentes. Est-ce que cela vous ennuie ? »

Ces questions sont vitales lors d'un projet externalisé, et cette ambiguïté autour de tels problèmes est source de conflits possibles – et de litiges. La clarté (autant que cela puisse être réellement possible) concernant le *degré de finition*, *l'acceptation*, et *la correction* à la fois dans l'esprit des parties et dans le texte du contrat, devrait être un souci majeur pour les professionnels du droit. Ces questions peuvent aider considérablement au désamorçage des pièges grâce à une attention extrême portée à la *négociation* d'un cadre contractuel d'acceptation encourageant la *collaboration*.

L'acceptation (finale NdT) existe encore, mais elle est grandement simplifiée avec la livraison et l'acceptation itérative, parce que l'acceptation est accordée de manière incrémentale et adaptative à chaque itération. De plus, les pratiques agiles incluent, généralement, des tests d'acceptation hautement automatisés afin qu'il y ait peu ou aucun effort manuel (humain) demandé pour la validation.

L'acceptation se bâtit sur elle-même de telle manière que l'acceptation finale est le point culminant des acceptations ayant eu lieu tout le long du cycle de vie du projet, idéalement pour la plupart vérifiées de manière répétitive avec les tests d'acceptation automatisés.

En termes de travail contractuel, cela signifie que la définition et la négociation de l'acceptation ne doivent pas être un exercice exhaustif massif, seul le *cadre pour l'acceptation* doit être contractuellement clair.

Plus largement, pour chaque itération, l'acceptation est basée sur la conformation à un jeu de test d'acceptation préalablement établi, et dans le cas de Scrum, en conformation avec la « définition du fini ».

Un autre élément d'acceptation dans le développement agile – qu'il convient de considérer dans le cadre contractuel – est d'inclure les futurs utilisateurs du nouveau système dans la définition de l'acceptation et des tests d'acceptation. Les professionnels du droit soucieux du succès d'un projet devraient demander, « Est-ce que les bonnes personnes – les utilisateurs finaux – impliqués dans l'acceptation, collaborent-ils avec le fournisseur ? Et ceci à chaque itération »

## Exemples de clauses

Contrairement à ce qui a été énoncé en introduction, nous avons décidé qu'un exemple dans ce cas de figure aiderait à clarifier cette suggestion :

- a) Client et fournisseur définissent l'acceptation des livrables comme suit :
  - i. Les livrables passent tous les nouveaux tests automatisés et manuels définis avant la dernière itération.
  - ii. Les livrables passent tous les tests automatisés et manuels antérieurs, vérifiant qu'aucune régression n'est apparue.
  - iii. Les livrables sont conformes à la « définition de fini » définie avant l'itération
- b) Les tests d'acceptations sont définis à chaque itération, de manière incrémentale conjointement entre le client et le fournisseur (« Groupe d'Acceptation »), y compris les futurs utilisateurs des livrables. Le groupe d'acceptation fait une revue d'acceptation à la fin de chaque itération, en commençant à partir de la revue de sprint.
- c) Le client aura une période correspondant à la moitié des jours ouvrés d'une itération (« Période d'Evaluation », « Demi-Itération ») après la mise à disposition du livrable final pour vérifier que le livrable ou une partie de celui-ci n'est pas déficient.
- d) Si le client notifie le fournisseur par écrit avant l'expiration de la période d'évaluation que le livrable ou une partie de celui-ci est déficient sur n'importe quel aspect matériel (une « non-conformité »), le fournisseur devra corriger cette non-conformité dans un délai raisonnable mais pas plus long que la longueur d'une itération, après quoi le client aura une période additionnelle de demi-itération (« période de vérification ») débutant à réception des livrables corrigés ou d'une partie de ceux-ci pour vérifier que la non-conformité spécifique a été corrigée.
- e) Le client fournira au fournisseur toute assistance raisonnable requise pour vérifier l'existence d'une non-conformité et la corriger.

## Limitation de responsabilité

La négociation des clauses de responsabilités est peut-être la chose la plus difficile lors de toutes négociations contractuelles, et une approche agile n'y changera rien. Néanmoins, elle peut aider. Par exemple, elle peut atténuer la responsabilité du fait d'avoir un livrable utilisable à la fin de chaque itération.

Par exemple, un défaut dans un livrable itératif peut avoir un impact moindre opérationnellement, la conséquence négative ayant été découverte plus tôt. Cela ne signifie pas qu'il n'y a pas de répercussions qui ne devront jamais être traités au travers du paradigme de responsabilité, mais les conséquences pourront être amoindries.

Considérons le cas que j'ai (Tom dans le cas présent) rencontré : dans un cycle de vie séquentiel d'un projet traditionnel pour un nouveau système de facturation, il a été découvert, après « une grosse livraison à la fin » que des montants dupliqués et erronés ont été envoyés à *de nombreux* clients. Les retombées et les coûts supplémentaires étaient considérables : l'entreprise a dû alléger certaines nouvelles factures, offrir des remises, et réparer son image auprès de ses clients – ainsi que

[www.agilecontracts.org](http://www.agilecontracts.org)

23/45

payer la correction des problèmes sous-jacents. Il y a eu ensuite un affrontement avec le fournisseur externe pour déterminer qui devait payer les dégâts – la responsabilité.

Dans une approche agile, ces mêmes factures problématiques pourraient aussi être envoyées. Mais il est aussi possible que ces factures auraient pu être envoyées plus tôt à un nombre moindre de clients, utilisant une version précoce du système avec juste les fonctionnalités essentielles pour tester sur le terrain cette fonctionnalité critique.

Ceci devrait réduire le coût, l'exposition, et les dégâts à la clientèle. Cela réduit aussi le coût de réparation parce que le système est plus petit avec moins d'adhérences entre ses composants logiciels.

Ainsi, la responsabilité peut être atténuée par le développement agile.

### Garantie

Identiques à la responsabilité, les intérêts liés à la garantie sont atténués par une approche incrémentale. Le profil de risque associé avec la garantie finale est considérablement amoindri grâce à la confiance et à l'acceptation du livrable lui même, grâce à l'acceptation incrémentale. Ceci est particulièrement vrai si les tests d'acceptation automatisés sont employés.

Comme pour la responsabilité, la garantie devrait être liée à chaque incrément du livrable opérationnel (à la fin de chaque itération) bien qu'il y ait toujours une garantie générale sur le produit final.

### Livrables

Les contrats de projets traditionnels incluent souvent une liste prescriptive, détaillée de ce qui devrait être livré (beaucoup de documents, ...), et comment l'acceptation de ces artefacts est réalisée. Quelques fois, ces détails sont inclus dans une annexe d'énoncés des travaux ou de « plan qualité ». Éviter d'inclure de telles spécificités et rigidités – éviter cette liste détaillée de livrable dans le contrat. Pourquoi ?

- Cela conduit à un accroissement d'activités inutiles au détriment de l'attention portée au développement d'un logiciel opérationnel, et il y a une présomption – possiblement infondée – que l'on sait quels artefacts ont de la valeur,
- Il y a une attention portée davantage sur la négociation et la conformation aux « plans qualité » plutôt que sur la coopération pour créer un logiciel utile.
- Cela renforce (l'illusoire) esprit de planning prédictif commande-contrôle plutôt que l'apprentissage et la réaction au changement.
- Cela renforce la croyance (infondée) qu'un système complètement défini peut être, de façon prévisible, commandé et livré comme s'il s'agissait d'un plat dans un restaurant plutôt que la découverte créative d'un travail.

Cela dit, nous avons vu un logiciel sur mesure dont le code source n'a jamais été fourni par le fournisseur – quelqu'un a dû oublier. Ainsi, il y a des cas dans lesquels le client ne comprend pas ce qui est critique. Mais dans de tels cas, la découverte des livrables apportant de la valeur peut être faite simplement à travers des livraisons et des déploiements incrémentaux, plutôt qu'à travers une



liste contractuelle de livrables.

A l'occasion, une documentation technique pour aider la maintenance est importante – elle est couramment utilisée comme une aide à l'apprentissage pour les personnes novices au système – et sa livraison est souvent spécifiée dans un contrat de projet traditionnel. Jusque-là, la maintenance d'un système déployé récemment est fréquemment externalisé aux mêmes personnes ayant créé le système et qui ont le moins besoin d'une telle documentation. Par conséquent, cela peut être un gâchis de l'exiger comme un livrable à avoir au plus tôt. Si, dans le futur, il y a un besoin *démontré* de documentation pour un client (par exemple, si le client prend en charge le travail de maintenance), alors elle pourra être créée par le fournisseur, peut être lors d'un atelier conjoint agile de documentation avec le client *après que le système soit achevé*.

### Echéancier de paiement

Le système le plus populaire est, peut être, de payer à chaque itération, une fois qu'il y a eu acceptation finale pour cette itération. Dans un cas simple, tel que des contrats basiques progressifs, le paiement est de 100 % du prix négocié de l'itération. Des plans de paiements plus complexes sont liés généralement à des plans globaux de tarifications de projets plus complexes. Par exemple, dans les divers systèmes « pertes/profits partagés » tels que les contrats coûts cibles, en plus des paiements par itération, il y a un paiement différé à la fin du projet. Ou, à chaque itération, il peut y avoir un pourcentage x retenu qui s'accumule et qui pourra être versé à divers jalons intermédiaires.

### Tarification

#### Temps et matériels (Régie)

Les variantes « Temps et matériels » (T&M) sont de bons modèles de tarification pour les projets agiles : simple, directs. A recommander. Noter que le T&M s'applique à la fois aux contrats à périmètre fixe ou variable.

Une problématique commune et traditionnelle sur les projets séquentiels de développement, en T&M, est que les clients sont bloqués dans un cycle sans fin de paiements avant d'en voir des résultats concrets. Une autre problématique classique est de savoir si les clients en ont pour leur argent. Ces problématiques sont améliorées par une approche agile avec un système utilisable à chaque itération – les progrès sont mesurés en terme de fonctionnalités logicielles utilisables, de grande transparence, et de la possibilité d'arrêter à la fin de n'importe quelle itération.

T&M demande de la confiance et de la transparence entre les parties. Cela demande du temps et des efforts sincères pour l'établir. Sur plusieurs projets, Valtech India a commencé avec des variantes de contrats à prix fixe, puis après avoir établi la confiance, a pu changer vers des variantes de modèles T&M avec leurs clients.

Plusieurs variantes du T&M limitent l'exposition du client et/ou équilibrent les pertes/profits. Par exemple :

- T&M plafonné (« ne pas excéder ») par itération
- T&M plafonné par projet ou livraison
- T&M plafonné par itération, avec ajustement – Pour l'itération suivante, le prix est plafonné,

mais si les objectifs initiaux de l'itération sont atteints (livrés) et acceptés, à un coût T&M en dessous du plafond initialement prévu, il y a paiement de la différence au fournisseur, tel que la moitié du montant économisé en dessous du plafond. Un plan similaire de tarification pertes/profits peut être utilisé dans un modèle à coût cible de niveau projet.

### Prix fixe par itération (par unité de temps)

Ce modèle a la vertu de la simplicité et de la prévisibilité, et il n'est pas inhabituel chez les prestataires externalisés. Il existe deux éléments clés :

- les exigences sont définies et négociées avant l'itération
- les exigences sont variables ou non définies a priori

Dans le premier cas, les problèmes sont identiques au modèle à prix fixe d'un grand projet... Le fournisseur doit clarifier le travail et avoir suffisamment confiance dans les estimations afin de ne pas perdre d'argent. Le périmètre réduit d'une itération rend cela plus faisable que pour un gros projet. Le problème clé (ou le coût) pour les clients est l'ajout par le fournisseur d'honoraires conditionnés au risque associé au caractère variable et imprévisible des travaux de recherche et développement.

Dans le deuxième cas, l'élément clé est la confiance du client envers le fournisseur. La transparence, les livraisons fréquentes et l'arrêt facile sont une aide à cela.

### Prix fixe par unité de travail

Plusieurs fournisseurs externalisés agiles offrent un modèle à prix fixe par unité d'œuvre (UdT). En contraste au développement traditionnel, où une UdT pourrait signifier un document ou une toute autre solution incomplète, une UdT agile reflète le 7ème principe agile : *un logiciel qui fonctionne est la première unité de mesure du progrès*. Ce qui signifie que l'UdT est liée à des fonctionnalités testées et opérationnelles.

Ces modèles se rencontrent sous différents noms avec différents systèmes d'estimation de l'UdT : « prix par story point », « prix par point de fonctions », « prix par point de fonctionnalité », et ainsi de suite.

*Points de taille contre valeur* – Dans les schémas, nous avons vu, le « point » est toujours lié à une estimation de la taille ou de l'effort – et ainsi, lié au *coût*. Même si les vendeurs proclament que le modèle de prix est lié à la *valeur* en utilisant une terminologie moderne à consonance agile tel que « prix par point », il est faux de dire qu'un « point de story » ou « point de fonctionnalité » est une mesure de la valeur – dans l'idiome de « bang for buck » (NdT : expression idiomatique signifiant en avoir pour son argent ou avoir quelque chose pour chaque euro dépensé/investi) un *point* reflète un *buck* pas un *bang*. Quoiqu'il en soit, il est en théorie possible de définir les points en terme d'*impact à valeur métier* – où ceci est mesuré en utilisant un système tel que les tables d'estimation d'impact de Tom Gilb (et il a d'ailleurs proposé un système de modèle de prix d'impact de valeur [Gilb05]) – mais nous ne connaissons aucune application de cette approche.

Nous avons vu les fournisseurs externalisés agiles utiliser l'un des deux schémas suivants pour déterminer le prix fixe par point : (1) une moyenne basée sur plusieurs projets précédents, et (2) un montant sur mesure. Dans ce dernier cas, le client paye au fournisseur, la valeur moyenne du point

pour quelques itérations (ou paye T&M ou ...) durant lesquelles le détail des coûts sera suivi. A partir de là le fournisseur et le client se mettent d'accord sur un prix fixe par point, basé sur ce coût plus une certaine marge de profit.

Ce modèle est congruent avec les thèmes lean et agile d'orientation-livraison et d'orientation-valeur : partant du principe qu'une UdT agile est vaguement liée à la valeur aux clients (ce qui n'est pas toujours vrai), ils paient proportionnellement à la valeur reçue. Toutefois, puisque dans les schémas que nous avons vus un point est lié à la taille ou à l'effort plus qu'à la valeur réelle perçue par le client, ce *point* est en quelque sorte perdu.

Un problème clé pour appliquer ce modèle – méritant une attention particulière dans le contrat – est un cadre clair et partagé (par le client et par le fournisseur) pour définir un *point*. Par exemple, seuls les points de fonctions sont définis quasiment sans ambiguïté – et peuvent être identifiés et vérifiés par des analystes point-fonctions certifiés. En contraste, les points de story (connus également comme les points d'efforts relatifs) n'ont pas de signification *indépendante*.

### Modèles Paiements-par-utilisation

XPLabs (une entreprise de développement agile en Italie) promeut les contrats paiements-par-utilisation avec ses clients. Chaque utilisation (habituellement, une transaction) d'un système de type progiciel ou réalisé sur mesure et qui est déployé chez un client est automatiquement tracée par XPLabs. Le client est régulièrement facturé, basé sur la fréquence d'utilisation – un modèle simple de facturation. Cette approche tend à faire correspondre les intérêts du client et du fournisseur, et chacune des parties est gagnante si le système est utilisé de manière croissante.

S'il s'agit d'un système de type progiciel, le modèle est particulièrement attractif pour les clients : ils n'ont pas de coûts de maintenance ou de mise à jour, les coûts supplémentaires proviennent des améliorations faites sur mesure. Chaque nouveau déploiement chez un client est basé sur le même modèle simple de contrat.

S'il s'agit d'un système sur mesure pour un seul client, le modèle de contrat pour la phase de développement peut être n'importe laquelle des autres approches discutées ici, telle que T&M, peut-être à un taux en-dessous de la moyenne pour anticiper le passage au mode paiement-par-utilisation.

### Modèles hybrides de pertes/profits partagés

Il existe plusieurs plans de paiements hybrides dans le monde des affaires, bien connus et non évoqués ici (cf. lectures recommandées). Un modèle de pertes/profits partagés applicable au développement agile est celui proposé par Bob Martin [Martin04] :

Prix fixe promotionnel par unité d'œuvre, plus T&M promotionnel – Par exemple, en partant du scénario de projet suivant :

Estimation du projet	Vélocité moyenne (140 personnes, Itérations de 2 semaines)	Estimation originelle en jours-hommes	Paiement avec l'hypothèse de \$500 par homme/jour
100 000 points	4 000 points	35 000	\$17 500 000

Dans ce modèle, un coût journalier plus bas est offert, avec un coût supplémentaire par unité de travail. Par exemple, en prenant un coût journalier de \$500. Le fournisseur offre un prix promotionnel journalier de \$150, et un prix promotionnel par point de \$122,50<sup>7</sup>:

Nombre de jours-hommes réels	Facturation client réelle	Variations entre l'estimation et l'effort réel	Variations entre l'estimation et la facturation réelle	Coût réel jours-hommes
30 000	\$16 750 000	-14%	-4%	\$558
35 000	\$17 500 000	0	0	\$500
40 000	\$18 250 000	+14%	+4%	\$456

#### Observations :

- Si l'effort réel est égal à l'estimation originale (35 000 jours/homme, 100 000 points), le paiement du client est égal simplement au plan T&M à \$500 par jours/homme.
- Si l'effort réel varie, le paiement du client varie moins sévèrement
- Comme avec le coût cible et quelques autres plans d'ajustement, le client et le fournisseur partage la perte ou le profit si le projet nécessite plus d'efforts que prévu lors l'estimation originale.

#### Prix fixe par projet et prix coût cible

Ces deux plans de tarifications sont détaillés dans la prochaine section. Bien au-delà de la simple tarification, leurs impacts s'étendent à la globalité du contrat ou au modèle de projet.

<sup>7</sup> Le prix de \$122.50 est calculé : Si l'on suppose un taux du jour.homme de \$150, et 35 000 jours hommes, et 100 000 points, c'est le prix par point nécessaire pour aboutir à un paiement total de \$17 500 000

## 3EME PARTIE : MODELES DE CONTRATS

Depuis l'aube des temps, les êtres humains ont rédigé des contrats, y mettant leurs espoirs et leurs peurs. Il existe une myriade de modèles et de variantes. Consulter la section des lectures recommandées pour en savoir plus. Cette section se concentre sur les modèles les plus communs et leurs variantes que les clients et fournisseurs rencontreront ou envisageront lors des projets agiles.

### Éviter ... les contrats à prix fixe, périmètre fixe (PFPF)

Les projets et les contrats à prix fixe, périmètre fixe – et encore pire durée fixe – tendent à conduire à des situations perdants-perdants à la fois pour le client et pour le fournisseur. Souvent, les clients n'obtiennent pas ce dont ils ont vraiment besoin, et les fournisseurs peuvent perdre facilement de l'argent. Et l'effort pour livrer dans la limite des contraintes de prix et de périmètre entraîne souvent les fournisseurs à dégrader la qualité de leur travail – en réduisant la qualité du code, en réalisant moins de test, et ainsi de suite. Tous ces éléments conduisent à un accroissement des coûts futurs pour les clients qui auront, éventuellement à payer pour les péchés du passé, en émettant par la suite des demandes de changements<sup>14</sup> pour obtenir ce dont ils ont réellement besoin avec des coûts de maintenances plus élevés pour un logiciel de moindre qualité ayant une forte « dette technique ».

Les offres à prix fixe ont ajouté une importante contingence de risques (un pourcentage à hauteur de 50 % du coût estimé) au prix global – cette prime est généralement dissimulée dans l'effort estimé<sup>15</sup>. Ceci conduit à une réduction de la transparence et à un accroissement du jeu durant l'exécution du projet, le fournisseur voulant cette prime pour réaliser un profit plutôt que pour consommer un budget. Qui plus est, étant donné que les spécifications préalables signées n'indiquent presque jamais ce qui est vraiment nécessaire (à cause d'une myriade de facteurs dus à la complexité inhérente de ce domaine), le fournisseur génère encore plus de revenus – en Inde, les fournisseurs externalisés appellent ceci des « rentes » – à travers une série de demandes de changement, chacune pour un coût additionnel au-delà du prix fixé à l'origine.

Les projets à prix fixe ont été mis en avant sous le couvert de divers facteurs d'optimisations locales (autres que le succès du projet) ; nous encourageons les professionnels du droit à prêter attention à celles-ci :

- En tant que client, le plus important est de connaître le coût, pour établir le rapport financier ou le budget<sup>16</sup>
- En tant que fournisseur, le plus important est de facturer la valeur totale commandée
- En tant que commercial, le plus important est de facturer la valeur totale commandée, pour

14 Les prestataires traditionnels externalisés en Inde apprécient ce modèle de demandes de changement, cela leur permet d'en tirer plus de profits ; ils savent très bien que leurs contrats PFPF ne permettent pas de livrer ce dont le client a vraiment besoin, et ils attendent et obtiennent cette rente (comme ils l'appellent en Inde) de l'évolution du système insatisfaisant pour répondre aux vrais besoins.

15 Pour ceci et d'autres raisons, les contrats PFPF sont aussi appelés contrat « le plus tard possible, le plus cher possible »

16 Voir la section Au-delà du budget au chapitre Organisation dans le livre compagnon, pour une alternative aux processus budgétaires traditionnels

obtenir la commission maximale

- En tant que manager, le plus important est d'éviter de perdre du temps sur le projet. Nous voulons commander quelque chose, retourner à d'autres tâches sans d'autres « interruptions » et être livrés à la fin.

Mais il y a des entreprises, pour une raison ou une autre, qui essayent encore. Dans ce cas, la question la plus fréquente que nous ayons est, « Comment faites-vous pour réaliser des projets à prix fixe et périmètre fixe en méthode agile ? ». Tout d'abord, c'est possible : Valtech India et d'autres prestataires externalisés l'ont fait (à cause de la demande du marché) – même s'il s'agit du modèle qu'ils apprécient le moins.

Il y a deux malentendus fréquents sous-jacents à la question du PFPF et des méthodes agiles :

- Le premier malentendu est que les exigences nécessaires à la livraison globale du projet ne sont pas connues ou estimées avant la fin de la première itération lorsqu'une méthode agile est utilisée. Faux. Plus exactement, en Scrum, avant l'itération 1, il peut y avoir une planification initiale de livraison (« création d'un product backlog initial ») dans lequel toutes les exigences identifiables sont clarifiées et estimées.
- Le deuxième malentendu est que les exigences doivent changer avec les méthodes agiles. Faux. Plus exactement, toutes les méthodes agiles offrent l'opportunité et les mécanismes d'apprentissage et de changement, sans les rendre obligatoire. Scrum peut être utilisé avec un backlog de livraison à contenu fixe – en offrant toujours des bénéfices, ceci grâce à de meilleurs retours plus fréquents sur les manières de travailler, les technologies, les résultats de tests, et des lots plus petits.

Avec Scrum ou toute autre approche, il y a des clés pour éviter les catastrophes lors de la prise en charge d'un projet PFPF :

- Appliquer la plus grande diligence en termes d'analyse préalable détaillée des exigences, au travers de définition des tests d'acceptations, une estimation habile de l'effort de toutes les exigences, le tout effectué par les personnes expérimentées.
- Ne permettre aucun changement d'exigence ou de périmètre, ou seulement permettre de nouvelles exigences en remplacement des exigences existantes si elles sont d'un niveau d'effort égal.
- Accroître la marge du prix du contrat, pour refléter le risque significatif inhérent dans le développement d'un logiciel PFPF – un domaine plein de découverte, variabilité et de mauvaises surprises.
- Employer des experts du domaine expérimentés et focalisés sur l'excellence technique.

Noter qu'une culture lean à long terme, d'avoir de bons ingénieurs et des responsables-formateurs, experts dans le sujet et coachant leur équipe, fournit un environnement pour faire monter en compétence les personnes nécessaires et pour réduire le risque des projets PFPF.

## Fréquence de règlements

La fréquence de règlement est généralement par itération, avec un reliquat à la fin (si le total des

règlements n'a pas été précédemment payé). Le montant par itération est un pourcentage fixé sur le prix global, basé soit sur l'estimation du nombre total d'itération ou si le projet est aussi fixé, sur la durée du nombre prédéfinis d'itérations.

### Flexibilité des projets PFPF en Scrum

En Scrum, lors d'un contrat PFPF, plusieurs domaines existent dans lesquels, il est possible d'accroître la flexibilité, avec de faibles risques : la capacité de ...

- remplacer des exigences existantes par de nouvelles d'un niveau d'effort équivalent
  - cette option de *remplacement* est importante à souligner
- changer l'ordre d'implémentation des exigences fixées
- améliorer la « définition du fini » à chaque itération

Schwaber<sup>17</sup> suggère également deux autres provisions sur le contrat

- les clients peuvent demander des livraisons supplémentaires à n'importe quel moment, tarifé avec T&M.
- Les clients peuvent résilier plus tôt s'ils sont satisfaits plus tôt, un règlement d'un montant de 20 % du montant non facturé est payé au fournisseur.

Les professionnels du droit doivent être conscients que cette flexibilité peut et doit être exprimée dans les clauses du contrat PFPF.

### Devrions-nous faire des projets PFPF avec une approche traditionnelle séquentielle ?

Une question qui est quelques fois posée est, « Si nous devons faire un projet PFPF, devrions-nous utiliser une méthode agile ou un cycle de vie séquentiel (cascade, ...) et une approche traditionnelle ? ».

Il est évident qu'un développement en cycle de vie séquentiel est corrélé avec un coût plus élevé, une livraison plus lente, une productivité plus faible, plus d'anomalies, ou des taux d'erreurs plus élevés, comparés avec des méthodes itératives, incrémentales ou agiles [MacCormack01, Reifer02, DBT05, MJ05, CSSD05, AV07, PRL07].

Et la dernière chose que vous voulez faire avec un projet PFPF est de l'empirer en appliquant une approche traditionnelle et séquentielle de développement.

C'est tout le contraire : si vous réalisez un projet PFPF en Scrum, vous aurez moins de gâchis, moins de files d'attente, moins de travail en cours, et vous gagnerez plus tôt des retours sur la vraie nature du projet. Basé sur des retours précoces, vous pourrez *ajuster précocement* au lieu de tardivement. Tout spécialement lors des projets PFPF, vous voulez savoir *où en sont réellement les mauvaises choses le plus rapidement possible* ; les méthodes agiles améliorent ces retours.

Il y a un avantage plus subtil à utiliser une approche agile dans un projet PFPF : il peut évoluer en un projet flexible orienté collaboration. Beaucoup de clients qui sont parties prenantes comprennent

---

<sup>17</sup> Ken Schwaber est le co-créateur de la méthode agile Scrum

que le modèle PFPF peut ne pas résoudre leurs problèmes, mais il leur est imposé – peut être par les services juridiques et financiers. Une fois que les clients ont commencé à interagir directement avec le fournisseur agile sur un projet ostensiblement fixé, à voir les livraisons rapides toutes les deux semaines d'une solution bien faite, à réaliser leur capacité à changer l'ordre d'implémentation de leurs exigences (et à les remplacer) dans les itérations, et à construire une confiance et une collaboration avec le fournisseur, « fixé » peut devenir « flexible », le client se détend, il voit les avantages d'une « collaboration avec le client plutôt qu'une négociation contractuelle » et par conséquent est d'accord pour prêter moins attention à la définition originale et plus d'attention au développement d'évolutions pour satisfaire leurs vrais besoins.



## Partie 3 : Modèle de contrat

### Evolution du contrat sur un grand projet agile

avec Greg Hutching

Un exemple d'évolution d'un modèle de contrat variable multi-phase fût un projet de 15 000 jours hommes d'une durée de 3 ans entre Valtech (India) et un client détaillant. Il y eut quatre phases contractuelles successives (et non pré-planifiées) en réponse à l'apprentissage et l'adaptation : (1) PFPF par projet, (2) progressif, T&M par itération, (3) progressif, prix fixe par unité de travail, et (4) progressif, T&M plafonné par itération.

- (1) *PFPF par projet* – Le client était uniquement familier des projets à cycle de vie séquentiel, et c'était la première collaboration avec Valtech – le degré de confiance était faible. Aussi, s'attendaient-ils et ont obtenu un contrat traditionnel PFPF basé sur le cycle de vie séquentiel, avec une date de fin négociée non contractuelle. En dépit d'efforts substantiels pour valider les spécifications et les estimations, beaucoup de choses – comme d'habitude – se révélèrent inconnues et mal comprises. Après la première année, les coûts excédaient le budget et la livraison en retard de plusieurs mois sur la date souhaitée. Grâce aux efforts de Valtech pour être transparent et réactif, une certaine confiance s'était développée, et par conséquent le client accepta de remplacer le contrat PFPF.
- (2) *Progressif, T&M avec bonus/malus* – Pendant la première année, le client fût progressivement familiarisé (par Valtech) aux principes du développement agile, et fût donc ouvert à de nouvelles façons de développer et à un nouveau contrat : un contrat progressif à périmètre variable basé sur Scrum, tarifé en T&M par itération, ajusté (à la demande du client) avec des clauses de bonus/malus pour (1) la qualité des livrables des itérations, et (2) la vélocité. Sans surprise ([Austin 96]) ces clauses ont subtilement affecté le comportement des développeurs (une réduction de la transparence, plus de jeux) pour éviter les « malus ». Les progrès (une fois en Scrum) furent beaucoup plus rapides, la confiance s'améliorera avec une livraison à chaque itération et une collaboration rapprochée avec le client. (*à suivre ...*)

Enfin, après un contrat PFPF initial accompli avec Scrum, le client peut vouloir utiliser l'un des modèles alternatifs de contrat agile lors de projets ultérieurs. Plusieurs prestataires agiles externalisés ont eu ces retours d'expériences positifs avec leurs clients.

## Essayer ... Contrat progressif à prix variable et périmètre variable

Dans leurs formes les plus pures, les contrats<sup>18</sup> progressifs impliquent un périmètre complètement flexible défini de manière adaptative à chaque nouvelle itération. Ces contrats sont de bons candidats pour des projets agiles.

(suite)

- (3) *Progressif, prix fixe par unité de travail* – dans cette troisième phase du contrat, la tarification changea pour le prix fixe par point de cas d'utilisation (PCU), parce que le client pensait qu'il obtiendrait plus de valeur pour son argent. De plus, l'élément du bonus/malus fût supprimé. Le PCU fut choisi, le client étant très familier des cas d'utilisations, et il voulut une estimation des unités de travail en relation avec les cas d'utilisation. Les règlements variaient à chaque itération, basés sur le nombre de PCU livrés. En outre, à ce stade, le client était à l'aise pour conduire les itérations et l'acceptation avec une définition itérative des tests d'acceptations. Ceci combiné avec de meilleurs tests d'intégration thèmes par thèmes permis de lisser et améliorer l'acceptation à chaque itération. Cette forme de contrat fût considérée par Valtech et le client comme offrant le plus de satisfaction. Néanmoins, la méthode de l'unité de travail (PCU) demanda plus d'analyse préalable que dans les autres formes – une déférlante de spécifications détaillées de cas d'utilisations survenait généralement deux itérations avant l'implémentation.
- (4) *(Phase de support) Progressif, T&M plafonné* – La forme et la phase finale du contrat furent établies après le déploiement du produit, pour le support et les améliorations mineures. Le budget de support client fût fixé (per annum), avec un périmètre de travail variable. Par conséquent, le T&M plafonné (par mois) était acceptable – simple à gérer, car un haut niveau de confiance avait été établi.

[Poppendieck05]. Il existe des contrats de services ou des contrats cadres « parachutes » définissant la relation principale et le plan de tarification par itération, mais sans définir le périmètre. Les contrats progressifs ne définissent pas le prix fixe total du projet – même si une des variantes propose le plafonnement du projet.

La résiliation (du contrat) pouvant arriver à la fin de n'importe quelle itération – avec un système opérationnel, l'exposition du client (aux risques NdT) est ainsi contrôlée. Si les deux parties sont satisfaites avec cette relation, les contrats progressifs projets peuvent continuer indéfiniment.

18 Connu aussi sous la dénomination, ou comme une variante de, *périmètre variable ouvert, incrémental ouvert, ou livraison indéfinie quantité infinie (LIQI)*

Généralement (mais ce n'est pas une obligation) avant chaque itération, le client et le fournisseur définissent les objectifs de la prochaine, accompagnés éventuellement de tests d'acceptations. Quelques fois – cela est récurrent chez Valtech – les objectifs pour l'itération N sont clarifiés pendant l'itération N-2.

Le prix par itération couvre toute la gamme de variations : prix fixe par itération, T&M par itération, et ainsi de suite.

*Variations* – Chez Valtech, un contrat à *prix plafonné et périmètre variable progressif* est chose commune, il y a un prix plafonné pour le projet global. Le prix par itération est pour n'importe quelle variante, telle que T&M. Également chose fréquente, le *contrat progressif à prix plafonné et périmètre variable* avec un backlog de livraison non contractuel, dans lequel les parties créent – *avant que le contrat ne soit rédigé* – un backlog des objectifs de livraison. Ce backlog est inclus en annexe du contrat. Néanmoins, il est convenu que *rien* dans le backlog d'origine n'est contractuel.

(C'est du Scrum classique.) Pourquoi créer un backlog de livraison non contractuel avant que le contrat ne soit rédigé ? Ce backlog est utilisé pour estimer le plafond global de la livraison et pour offrir un point de départ pour une vision commune. Il est aussi fréquent de créer un backlog de livraison non contractuel dans une phase contractuelle antérieure.

Les contrats progressifs sont un modèle habituel entre les fournisseurs externalisés et les clients de longue date entre lesquels une relation de confiance s'est établie. Un modèle fréquent (il ne s'agit pas d'une recommandation) est :

1. les premiers contrats sont des variantes à périmètre fixe et prix fixe
2. plus tard, une bascule vers des contrats progressifs avec un simple T&M ou un T&M plafonné par itération

### Essayer ... Accroître la flexibilité dans le projet et les variables du contrat

Un contrat progressif pur, à périmètre variable, prix variable et date variable est flexible. N'importe quelle variable (périmètre, prix, date) peut varier en flexibilité, dépendant du niveau de confiance et collaboration entre le client et le fournisseur – ou autrement limité, par une législation existante. Les variantes de contrat que les fournisseurs externalisés ont créées comprennent :

**Prix plafonné, <sup>19</sup>périmètre variable** – Discuté à la section précédente

**Prix plafonné, périmètre partiellement variable** – Un ensemble réduit d'exigences est fixé – laissant ainsi de la place pour l'apprentissage et l'adaptation.

**Prix fixe, périmètre variable** – Le contrat *périmètre optionnel* [BC99] est une variante de ce modèle. Il fixe également la date de fin.

### Le risque limité dans les contrats flexibles : le modèle multi-phase

Le modèle multi-phase est décrit dans la section p39 « Essayer ... les cadres du modèle variable multi-phase ». En bref, il met en œuvre un projet plus long à partir d'une série de contrats plus courts.

---

<sup>19</sup> Dans cette section, le prix se réfère au prix du projet global

Si la confiance n'est pas au rendez-vous, les clients peuvent limiter leurs risques (et leurs peurs) en utilisant des séries de contrats de courte durée et flexibles. Par exemple, un contrat *d'un an*, prix fixe, date fixe, périmètre variable peut être perçu avec une certaine appréhension. Mais une série de contrats de deux mois, prix fixe, date fixe, périmètre variable – avec la capacité de résilier à la fin de n'importe quel cycle – est plus savoureux.

En plus après quelques cycles de contrat, la confiance peut s'établir. A ce point, le client peut se diriger vers un simple contrat progressif avec un prix T&M par itération.

### **Des modèles partageant ou déplaçant de manière adaptative les pertes et profits ou les risques**

Dans un projet, il y a des risques et des avantages potentiels – pour les deux parties. Et cela peut évoluer avec le temps. Par exemple, un projet PFPF semble faire supporter plus de risques au fournisseur, même si cela est une illusion pour les raisons expliquées précédemment.

Certains cadres, étudiés dans les paragraphes suivants, ont été spécifiquement élaborés pour partager ces risques et ces avantages et pour faire supporter les risques à la partie en mesure de pouvoir les gérer<sup>20</sup>.

Dans le meilleur des cas, ces cadres engendrent un alignement accru des motivations pour les parties car « elles y jouent leurs peaux ». Et ils peuvent améliorer fondamentalement l'honnêteté et instaurer de bonnes relations. Cette philosophie est au cœur du concept de l'approche gagnant-gagnant, et crée la confiance et les relations favorisant un climat d'affaires.

Mais les contrats ne créent pas eux-mêmes la confiance et l'alignement. Dans le pire des cas, ce genre de contrat est détourné dans un jeu de rejet de responsabilité, pour faire supporter les pertes à l'autre partie et gagner égoïstement.

Ces cadres incluent :

- le coût cible
- le modèle variable multi-phase
- le partage des profits

### **Essayer ... les contrats à coûts cibles**

Les contrats à coûts cibles peuvent aider à aligner les motivations des deux parties. Ils sont utilisés par Toyota avec ses fournisseurs, reflétant ainsi le pilier de la pensée lean sur *le respect des individus*, dans lequel Toyota essaye d'établir des relations stables à long terme avec ses fournisseurs basées sur la confiance et le respect mutuel.

Ce modèle présume une étape de planification de livraison initiale dans laquelle le périmètre général du projet est identifié. C'est une partie de *l'étape Une* pour établir le coût cible :

---

20 Cela signifie généralement mettre les risques liés aux exigences (le « quoi ») dans les mains du client, et de mettre les risques liés à la technique et à l'implémentation (le « comment ») dans les mains du fournisseur.

1. Toutes les exigences du projet sont identifiées, analysées, estimées en collaboration entre client et fournisseur.
2. Le coût du changement ou de l'accroissement du périmètre sont estimés en collaboration lors du projet. C'est important : les contrats à coûts cibles doivent prendre en compte l'effort et le coût global de manière réaliste autant que possible.
3. A partir de ces deux éléments, établir le coût cible.
4. Calculer le profit cible, basé sur le coût cible (par exemple, 15 % du coût cible).
5. Partager les détails et les résultats avec le client (c'est également important).

Pendant l'étape une, les clés du succès de ce modèle sont (1) un coût cible généré avec le plus grand soin et la plus grande compétence, du mieux possible, sans se faire d'illusion (2) une transparence des coûts (du fournisseur), afin que le client puisse prendre connaissance de tous les détails ayant conduit au résultat du calcul du coût cible.

L'étape numéro deux dans un contrat à coût cible est l'exécution du projet – par exemple, en Scrum. Dès que cela est décidé, une pratique essentielle pour réussir est le suivi de tous les *coûts réels* dès qu'ils se produisent (par exemple, temps passé du développeur, réunions, matériels), et *partager* les informations de manière *transparente* avec le client en quasi-temps réel.

L'aspect clé des contrats à coûts cibles est une formule partagée des pertes et profits pour un équilibre résultant de la différence entre le coût réel et le coût cible. Il existe plusieurs variantes de cette formule.

Dans le cas le plus simple, un exemple pourrait être :

Ajustement = (Coût réel – Coût cible) \* Part du client pour le coût de la différence

Règlement du client = Coût cible + Profit Cible + Ajustement

Comme nous le verrons, *l'ajustement* peut être positif ou négatif.

En supposant que l'accord prévoit une répartition du coût de la différence de 60% pour le client, et 40 % pour le fournisseur. Cela donne :

Coût cible	Profit cible	Règlement client cible	Coût fournisseur réel	Ajustement	Règlement client réel	Profit fournisseur réel
1 000 000	150 000	1 150 000	1 100 000	+ 60 000	1 210 000	110 000
1 000 000	150 000	1 150 000	900 000	- 60 000	1 090 000	190 000

Si les coûts sont plus élevés que l'estimation, le client et le fournisseur partagent les pertes : Le profit du fournisseur est plus faible et le client supporte une partie du poids du coût. S'il y a une économie sur les coûts, les deux parties partagent les profits : Le profit du fournisseur est plus élevé et le client paye moins que le coût cible prévu initialement.

Ceci implique que les deux parties peuvent – sans garantie – promouvoir pro-activement d'autres façons de réduire le gâchis pendant le projet.

### Variations sur le plan de règlements

Les rédacteurs de contrats ont donné naissance à une myriade de variantes incluant :

- un paiement client plafonné (paiement minimum).
- un taux fournisseur réduit si le coût cible est dépassé.

### Coût cible et profit cible ajustables

Un autre élément essentiel des contrats à coût cible, soutenant les valeurs agiles et itératives, est la capacité à ajuster le coût cible (et le profit) par des négociations continues entre les parties. Les clés du succès de l'ajustement du coût cible peuvent être :

- un suivi comptable ouvert du projet réalisé par le fournisseur, d'une grande transparence, en quasi temps réel afin que le client puisse voir l'état réel des dépenses engagées par le fournisseur.
- une volonté de travailler ensemble partagée à la fois par le client et le fournisseur pour s'améliorer continuellement
  - c'est ce à quoi Toyota travaille dur [ISV09].
- un accord entre les parties, exprimé dès le début dans le contrat, sur les principes directeurs de l'ajustement du coût cible.
  - un cycle d'ajustement *modéré*, pour éviter d'être hyper réactif ou d'utiliser excessivement l'ajustement ; par exemple, une fois par itération est peut-être trop fréquent.

- le développement piloté par les tests d'acceptation, pour réduire l'ambiguïté.

Ces pratiques réduisent mais n'éliminent pas les discordes durant les renégociations continues, les problèmes d'ajustement étant souvent par nature confus – variation sur le thème « est-ce que c'est une anomalie ou une fonctionnalité ? ».

Un groupe reporta que le plan préétabli de classification des ajustements ci-après, a réduit les tensions :

Type	Description	Peut ajuster le coût cible ?
Réparation	Changement d'un besoin implémenté, à cause du fournisseur qui n'a pas fait ce qui aurait dû être « raisonnablement » compris ou fait	Non
Clarification	Changement d'un besoin « correctement » implémenté, à cause de la connaissance accumulée du client par retour d'informations	Non
Amélioration	Nouvelle fonctionnalité	Oui

Nous ne recommandons pas spécifiquement tel ou tel plan. En effet, ceci pouvant aussi bien aider que conduire à de longues et inutiles négociations, au lieu d'inciter à la coopération.

### Essayer... Cadres pour modèle variable multi-phase

Les projets comportent des incertitudes ou des profils de risque fluctuant avec le temps, qui se réduisent idéalement – pour les deux parties. Ceci peut être reflété par des cadres multi-phases qui mettent en avant ces profils changeants pour le client et le fournisseur. N'importe quelle phase peut utiliser n'importe quel modèle : PFPP, progressif, à coût cible, et ainsi de suite.

Un exemple de modèle variable multi-phases de chez Valtech reflète le modèle Scrum : (1) création du backlog initial (2) développement itératif adaptatif. Ce modèle fût utilisé sur un gros projet de B2B impliquant des parties prenantes dans 23 pays :

1. Phase 1 – Prix Fixe, durée fixe, périmètre variable. Il s'agissait, essentiellement, de la création du product backlog initial. Le livrable de cette phase était le product backlog – plus précisément, un backlog de livraison – et plusieurs documents d'analyse métier (analyse de marché, vision, ...), basés sur des ateliers et des analyses de l'équipe. Même si une liste spécifique de documents à livrer est identifiée dans le contrat, le périmètre de l'analyse ou le contenu variaient.

Avec un backlog de livraison, une estimation du coût est devenue possible, donc...

2. Phase 2 – Contrat progressif, T&M par itération, plafond de livraison, Backlog de livraison

non contractuel, durée fixe. La phase deux est essentiellement un projet classique Scrum : aucun élément dans le backlog original n'est lié au contrat, par contre, il sert à estimer, à fixer le coût de la livraison du projet, à offrir une vision initiale, et un point de départ sur ce qu'il y a à faire pour la première itération.

Pourquoi s'embêter avec ces modèles multi-phases plutôt que de simples contrats progressifs ? Typiquement, la motivation est (1) le manque de confiance, (2) une contrainte réglementaire, (3) la croyance par l'une des parties, qu'elle peut faire ou économiser plus d'argent ou réduire plus les risques, (4) un besoin de définir la vision, des exigences de haut niveau, ou le coût d'une phase ultérieure (et l'effort de ce travail lui-même est conséquent), ou (5) pour « optimiser » un objectif secondaire (autre que le succès du projet) tel que la meilleure prévisibilité des coûts.

Un autre exemple [Larman03]:

1. Phase une – prix fixe périmètre partiellement fixé pendant trois itérations (durée fixe). Le périmètre fixé obligatoire est « restreint » (par exemple, 20 % de l'effort disponible), laissant ainsi beaucoup d'espace pour la variabilité et la découverte lorsque les incertitudes sont grandes. Les risques du client et du fournisseur sont traités, et les deux parties apprennent beaucoup de la nature du projet qui se poursuit pour la phase deux.
2. Phase deux – PFPF, durée variable. Les risques pris par le fournisseur pour une phase à PFPF sont réduits grâce à la connaissance acquise et à la réduction de certaines sources de variabilité durant la phase une.

## CONCLUSION

« Comment pouvons-nous faire du développement agile lorsque des contrats sont impliqués ? » C'est une des questions à laquelle nous devons souvent répondre. Mais les problèmes clés ne sont pas avec le contrat, ils sont avec les rédacteurs du contrat et les clients qu'ils servent – dont la croyance est que la clé du succès tourne plus autour de la négociation du contrat et moins autour de la collaboration avec le client, ou que le succès du projet n'est pas l'objectif du travail du contrat. Ceci dit, pour réitérer un aphorisme de la pensée systémique, « il n'y a pas de blâme », ce qui implique que le comportement des personnes à l'intérieur du système est conditionné par lui – dans le cas présent par l'encouragement des silos des départements, des cibles locales (et des récompenses locales) conduisant à des optimisations locales, et le message subtile, « les avocats n'ont pas besoin d'apprendre les détails opérationnels ou des nouvelles approches de R&D, c'est le boulot de quelqu'un d'autre. »

De plus, lorsque cette question est posée, le mot « contrat » est souvent utilisé comme synonyme de contrat à « prix fixe et périmètre fixe », ce qui bien sûr, n'est pas du tout nécessaire – il y a une grande variété de modèles de contrat, y compris les contrats à périmètre variable et progressif, et bien d'autres encore.

Les professionnels du droit ont le devoir de considérer les ramifications des conséquences de la rupture de confiance et de collaboration – ainsi que d'autres problèmes – lorsqu'ils élaborent un contrat. Tout comme un avocat du droit des affaires a besoin d'apprendre plus sur les principes lean et agiles, les autres parties ont besoin d'en apprendre plus sur les nécessaires et légitimes



préoccupations des avocats. Et tout comme la fabrication d'un produit s'améliore du fait d'équipes de développement pluridisciplinaires, d'autres améliorations sont possibles en impliquant les professionnels du droit dans des équipes pluridisciplinaires encore plus larges.

## **LECTURE RECOMMANDEES**

Une première étape est de lire les suggestions listées dans la section « Essayer ... les avocats étudient les concepts du système de pensée agile et itératifs » en page 4.

Plusieurs sources existent, la plupart sur la Toile, sur le développement agile et les contrats ; néanmoins certaines sont plus théoriques que basées sur l'expérience – garder ceci à l'esprit lorsque vous les lirez. Voici quelques possibilités :

? Ces dernières années, Mary et Tom Poppendieck, chefs de file du développement lean de logiciels, ont organisé des ateliers d'élaboration de contrats, ont collecté et ont partagé des articles et des présentations sur leur site [www.poppendieck.com](http://www.poppendieck.com). Suivant un thème similaire à celui de ce chapitre, les propres travaux des Poppendieck sur les contrats agiles mettent l'accent sur les problèmes sous-jacents de confiance, collaboration et transparence liés aux contrats.

? Susan Atkinson et Gabrielle Benefield ont écrit sur le sujet des contrats agiles sous l'appellation « Modèle de contrats évolutifs » ; leur article est consultable sur [www.infoq.com](http://www.infoq.com).

? Sur [agilesoftwaredevelopment.com](http://agilesoftwaredevelopment.com) Peter Stevens a écrit un article résumant les « 10 contrats pour votre prochain projet agile ».

? Barry Boehm et ses collègues ont écrit plusieurs articles (disponible sur la Toile) sur le « modèle d'engagement incrémental ».

? Certaines personnes nouvelles sur le sujet pensent que les contrats encourageant la flexibilité, la collaboration, et l'alignement des intérêts (contrats « agile ») sont un concept nouveau, mais en réalité, beaucoup de choses ont été écrites et mises en avant ces dernières années, y compris par le gouvernement des États-Unis (voir par exemple *Administration des contrats gouvernementaux*). Il existe des dizaines, si ce n'est des centaines de livres et de sites web discutant de la diversité des modèles de contrats.

? Nous avons revus plusieurs modèles de contrats « publics » existants et portant explicitement sur le développement itératif, évolutionnaire ou agile. Cependant, Valtech et ThoughtWorks – et d'autres prestataires agiles que nous connaissons – ont écrit leurs propres contrats plutôt que d'utiliser ces modèles. Nous décourageons l'écriture de contrat « copier-coller », mais les modèles de contrats ci-dessous sont une source d'inspiration :

- Le consortium DSDM (DSDM est une méthode agile) propose un exemple de contrat, mis à disposition de ses membres sur [www.dsdm.org](http://www.dsdm.org). A noter que ce contrat est revu de temps à autre.
- Le contrat norvégien PS 2000 a été créé pour le développement itératif et évolutionnaire, par une alliance formée entre l'industrie et le gouvernement, disponible sur [www.dataforeningen.no](http://www.dataforeningen.no). Nous citons, « *[Le] contrat est conçu pour être utilisé lorsque c'est particulièrement difficile ou impossible d'élaborer une spécification détaillée avant de faire l'appel d'offre, l'idée étant de laisser au développeur de trouver la meilleure façon d'atteindre les objectifs et de satisfaire les besoins du client.* »

## REFERENCES

- [Austin96] Austin, R., 1996. *Measuring and Managing Performance in Organizations*, Dorset House
- [AV07] APLN, Version One, 2007. “2nd Annual Survey. The State of Agile,” VersionOne website at [http://www.versionone.com/pdf/stateofagiledevelopment2\\_fulldatareport.pdf](http://www.versionone.com/pdf/stateofagiledevelopment2_fulldatareport.pdf)
- [BC99] Beck, K., Cleal, D., 1999. “Optional Scope Contracts,” at [www.jarn.com/about/OptionalScopeContracts.pdf](http://www.jarn.com/about/OptionalScopeContracts.pdf)
- [CSSD05] Ceschi, M., Sillitti, A., Succi, G., De Panfilis, S., 2005. “Project Management in Plan-Based and Agile Companies,” *IEEE Software*, May/June 2005
- [DBT05] Dalcher, D., Benediktsson, O., Thorbergsson, H., 2005. “Development Life Cycle Management: A Multiproject Experiment.” *Proceedings of the 12th International Conference on Engineering of Computer-Based Systems*, IEEE Computer Society
- [EMH05] Eckfeldt, B., Madden, R., Horowitz, J., 2005. “Selling Agile: Target-Cost Contracts.” *Proceedings of Agile 2005 Conference*
- [Gilb05] Gilb, T., 2005. *Competitive Engineering*, Butterworth-Heinemann
- [Herzberg87] Herzberg, F., 1987. “One More Time: How Do You Motivate Employees?” *Harvard Business Review*, Sept/Oct 1987
- [ISV09] Iyer, A., Seshadri, S., Vasher, R., 2009. *Toyota’s Supply Chain Management: A Strategic Approach to Toyota’s Renowned System*, McGraw-Hill
- [Kohn93] Kohn, A., 1993. *Punished by Rewards*, Houghton Mifflin
- [Larman03] Larman, C., 2003. *Agile and Iterative Development: A Manager's Guide*, Addison-Wesley
- [LV08] Larman, C., Vodde, B., 2008. *Scaling Lean & Agile Development: Thinking and Organizational Tools for Large-Scale Scrum*, Addison-Wesley
- [MacCormack01] MacCormack, A., 2001. “Product-Development Practices That Work,” *MIT Sloan Management Review*. Vol. 42, No. 2.
- [Martin04] Martin, R., 2004. “Estimating Costs Up Front,” *Extreme Programming mailing list*, at <http://groups.google.com/group/comp.software.extreme-programming/msg/9a203fad85f3d363?hl=en>
- [MJ05] Moløkken-Østvold, K., Jørgensen, M., 2005. “A Comparison of Software Project Overruns—Flexible versus Sequential Development Models,” *IEEE Transactions on Software Engineering*, Vol. 31, No. 9, Sept 2005
- [Moore91] Moore, G., 1991. *Crossing the Chasm*, HarperCollins Publishers
- [Parkinson57] Parkinson, C., 1957. *Parkinson’s Law*, Buccaneer Books
- [Poppendieck05] Poppendieck, M., 2005. “Agile Contracts” *Agile 2005 Conference Workshop*, at [www.agilecontracts.org](http://www.agilecontracts.org)

[www.poppendieck.com/pdfs/AgileContracts.pdf](http://www.poppendieck.com/pdfs/AgileContracts.pdf)

[PRL07] Parsons, D., Ryu, H., Lal, R., 2007. “The Impact of Methods and Techniques on Outcomes from Agile Software Development Projects,” IFIP—Organizational Dynamics of Technology-Based Innovation: Diversifying the Research Agenda. Springer (draft)

[PS06] Pfeffer, J., Sutton, R., 2006. Hard Facts, Dangerous Half-Truths And Total Nonsense, Harvard Business School Press

[Reifer02] Reifer, D., 2002. “How Good are Agile Methods?” IEEE Software, July/Aug 2002.

[Smith07] Smith, P., 2007. Flexible Product Development: Building Agility for Changing Markets, Jossey-Bass

## BIO DES TRADUCTEURS

### Laurent CARBONNAUX



Expert Agile chez [Valtech](http://www.valtech.fr) (<http://www.valtech.fr>) depuis 2006. J'aide et accompagne les entreprises à transformer leur mode de production pour s'adapter aux besoins mouvants du marché. J'appuie mon expertise Agile sur mes multiples expériences techniques et d'architecture. J'ai eu la chance d'être certifié CSM par Craig Larman. Et bien entendu, fervent défenseur de l'agilité au sein de la [SigmaT](http://sigmat.fr/) (<http://sigmat.fr/>) et l'a-cube « Association pour l'Agilité en Auvergne ». Vous pouvez me retrouver sur mon [Blog](http://laurent.carbonnaux.org/) (<http://laurent.carbonnaux.org/>), ainsi que sur [Twitter](https://twitter.com/#!/lcarbonnaux) (<https://twitter.com/#!/lcarbonnaux>).

### Nicolas MEREAX

Pendant des années et encore maintenant, je me suis posé les questions « comment ça marche ça ? », « et si on change le truc là, ça fait quoi ? », puis un jour je suis tombé sur l'agilité (aïe) ou plutôt elle m'est tombée dessus (ouille) depuis j'ai rajouté les questions « pourquoi devons-nous toujours faire comme ça, si ça ne fonctionne pas ou alors si mal que personne n'est content ? », et par conséquent « et pourquoi pas, alors pour de vrai, faire de l'agilité pour faire avancer les choses en respectant les personnes qui le font ? ». Depuis 2008, c'est ce que j'essaye de faire et de montrer aux autres ; je ne peux plus m'en passer !

### Pierre FAUVEL



Agile depuis 2005 (ingénierie, puis scrum master, puis product owner, puis coach pour Alcyonix (<http://www.alcyonix.com>)). J'aide quand je le peux mes amis du Club Agile Rhone Alpes de Lyon (<http://lyon.clubagilerhonealpes.org/>). Je vois maintenant l'agilité comme partie d'un tout (eXtreme Programming, Scrum, Lean / Kanban, Lean Startup, ...) visant à améliorer le fonctionnement des entreprises de la meilleure façon possible. J'avais travaillé sur la question des projets agiles au forfait, sujet épineux, et j'ai d'autant plus apprécié ce texte. Vous pouvez me retrouver sur mon blog (<http://pierrefauvel.wordpress.com>) et sur twitter ([https://twitter.com/#!/pierre\\_fauvel](https://twitter.com/#!/pierre_fauvel))