

Search

Menu

- [À propos](#)
- [Votre carrière](#)
- [Orga. personnelle](#)
- [Orga. d'équipe](#)
- [Outils](#)
- [Création d'entreprise](#)
- [Événements](#)
- [Technique](#)

Blogroll

- [43 Folders](#)
- [À propos de la création d'entreprise](#)
- [A smart bear](#)
- [Ask the wizard](#)
- [Créer une micro-startup](#)
- [Des livres à lire](#)
- [Éclaireur.net](#)
- [Entreprise 2.0](#)
- [FineInfo](#)
- [Gestion et versions des sources](#)
- [Jester Management](#)
- [Joel on software](#)
- [L'exemple de Gordon Ramsay](#)
- [Le triangle Qualité, Coût, Délai](#)
- [Les clés de la réussite](#)
- [Les connaissances de base](#)
- [Les erreurs à ne pas faire en entretien](#)
- [Les listes](#)
- [Les méthodes agiles concrètement](#)
- [Les réunions](#)
- [Les spécifications à problème](#)
- [onPK](#)
- [Rands in repose](#)
- [Réduire les goulots d'étranglement](#)
- [Signal vs. Noise](#)

- [Synergeek](#)
- [Web Worker Daily](#)

Blogroll

- [43 Folders](#)
- [À propos de la création d'entreprise](#)
- [A smart bear](#)
- [Ask the wizard](#)
- [Créer une micro-startup](#)
- [Des livres à lire](#)
- [Éclaireur.net](#)
- [Entreprise 2.0](#)
- [FineInfo](#)
- [Gestion et versions des sources](#)
- [Jester Management](#)
- [Joel on software](#)
- [L'exemple de Gordon Ramsay](#)
- [Le triangle Qualité, Coût, Délai](#)
- [Les clés de la réussite](#)
- [Les connaissances de base](#)
- [Les erreurs à ne pas faire en entretien](#)
- [Les listes](#)
- [Les méthodes agiles concrètement](#)
- [Les réunions](#)
- [Les spécifications à problème](#)
- [onPK](#)
- [Rands in repose](#)
- [Réduire les goulots d'étranglement](#)
- [Signal vs. Noise](#)
- [Synergeek](#)
- [Web Worker Daily](#)

Blogroll

- [43 Folders](#)
- [À propos de la création d'entreprise](#)
- [A smart bear](#)
- [Ask the wizard](#)
- [Créer une micro-startup](#)
- [Des livres à lire](#)
- [Éclaireur.net](#)
- [Entreprise 2.0](#)
- [FineInfo](#)

- [Gestion et versions des sources](#)
- [Jester Management](#)
- [Joel on software](#)
- [L'exemple de Gordon Ramsay](#)
- [Le triangle Qualité, Coût, Délai](#)
- [Les clés de la réussite](#)
- [Les connaissances de base](#)
- [Les erreurs à ne pas faire en entretien](#)
- [Les listes](#)
- [Les méthodes agiles concrètement](#)
- [Les réunions](#)
- [Les spécifications à problème](#)
- [onPK](#)
- [Rands in repose](#)
- [Réduire les goulots d'étranglement](#)
- [Signal vs. Noise](#)
- [Synergeek](#)
- [Web Worker Daily](#)

Blogroll

- [43 Folders](#)
- [À propos de la création d'entreprise](#)
- [A smart bear](#)
- [Ask the wizard](#)
- [Créer une micro-startup](#)
- [Des livres à lire](#)
- [Éclaireur.net](#)
- [Entreprise 2.0](#)
- [FineInfo](#)
- [Gestion et versions des sources](#)
- [Jester Management](#)
- [Joel on software](#)
- [L'exemple de Gordon Ramsay](#)
- [Le triangle Qualité, Coût, Délai](#)
- [Les clés de la réussite](#)
- [Les connaissances de base](#)
- [Les erreurs à ne pas faire en entretien](#)
- [Les listes](#)
- [Les méthodes agiles concrètement](#)
- [Les réunions](#)
- [Les spécifications à problème](#)
- [onPK](#)
- [Rands in repose](#)

- [Réduire les goulots d'étranglement](#)
- [Signal vs. Noise](#)
- [Synergeek](#)
- [Web Worker Daily](#)

Méta

- [Connexion](#)
- [Flux RSS des articles](#)
- [RSS des commentaires](#)
- [WordPress.org](#)



Les spécifications à problème (2) Spécifications qui changent en cours de développement

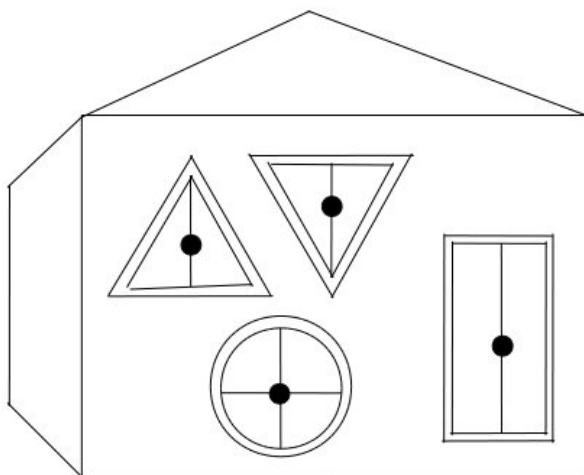
juin 8, 2011 at 9:42 by Amaury

Tags: [développement](#), [spécification](#)

Ce billet fait partie d'une série de trois articles consacrés aux spécifications et à leurs problèmes. Le précédent portait sur [les développements réalisés sans spécification](#). Je vous invite à le lire rapidement pour comprendre le contexte. Le troisième article porte sur [les spécifications qui arrivent après le développement](#).

En image

Spécifications qui changent en cours de développement



Explication de l'architecte : « Pour les fenêtres ? Ah oui, à force de changer d'avis... »

Version texte

On peut voir une maison dont la façade est couverte de fenêtres de tailles et de formes différentes.

Explication de l'architecte : « *Pour les fenêtres ? Ah oui, à force de changer d'avis...* »

Mon avis

Il s'agit là d'un cas d'école tellement cela arrive souvent. Une spécification fonctionnelle a été écrite, elle a été analysée (possiblement en faisant plusieurs itérations), une spécification technique en a découlé, et le développement a été commencé. Soudain, sous prétexte de vouloir apporter des « précisions » à la spécification, la spécification initiale est modifiée de manière subtile. Enfin, la subtilité est fonctionnelle ; d'un point de vue technique, les modifications nécessaires sont assez profondes. Mais si vous commencez à vouloir négocier, on vous rétorque que les modifications sont mineures, et que vous devez pouvoir vous adapter, que diable ! Sous-entendu : l'équipe technique doit prouver sa valeur en étant capable de gérer ces changements sans que cela n'affecte son planning.

Le problème, c'est que ce genre de chose implique quasi-automatiquement de prendre des raccourcis, afin de tenir les délais initialement prévus. L'une des conséquences directe est que l'environnement technique global (code source, schéma de base de données, documentation technique, etc.) se retrouve truffé de scories, des reliquats des différentes directions techniques qui ont été suivies puis abandonnées, à cause de tous ces pseudo-refactoring qui n'ont pas été prévus. Pour finir, vous obtenez des applications difficiles à maintenir et à faire évoluer.

Pourtant, il faut être fondamentalement prêt à faire face aux changements. La solution a été trouvée avec l'utilisation des [cycles itératifs](#) et des [méthodes agiles](#). Chaque itération permet de s'assurer que les développements vont dans le bon sens. Mais aucun changement de direction ne peut intervenir en cours d'itération.

Leave a Comment

Laisser un commentaire

Votre adresse de messagerie ne sera pas publiée. Les champs obligatoires sont indiqués avec *

Commentaire

Nom *

Adresse de contact *

Site web

Laisser un commentaire



Notifiez-moi des commentaires à venir via email. Vous pouvez aussi [vous abonner](#) sans commenter.

[View Full Site](#)