

Un Bureau en Mode eXtreme Programming

Emmanuel CHENU

emmanuel.chenu@fr.thalesgroup.com

<http://emmanuelchenu.blogspot.com>

Nous développons des logiciels temps-réel critiques, embarqués pour l'avionique.

Une nécessité de travailler en équipe

La complexité des besoins des clients, le volume des développements à réaliser et la rigueur imposée par les exigences de sécurité ne peuvent être appréhendés que par un travail d'équipe.

C'est en confrontant les points de vues, en partageant les expériences et en parallélisant certains travaux que nous pouvons développer des logiciels de qualité dans les délais attendus.

Un environnement physique adapté au travail en équipe

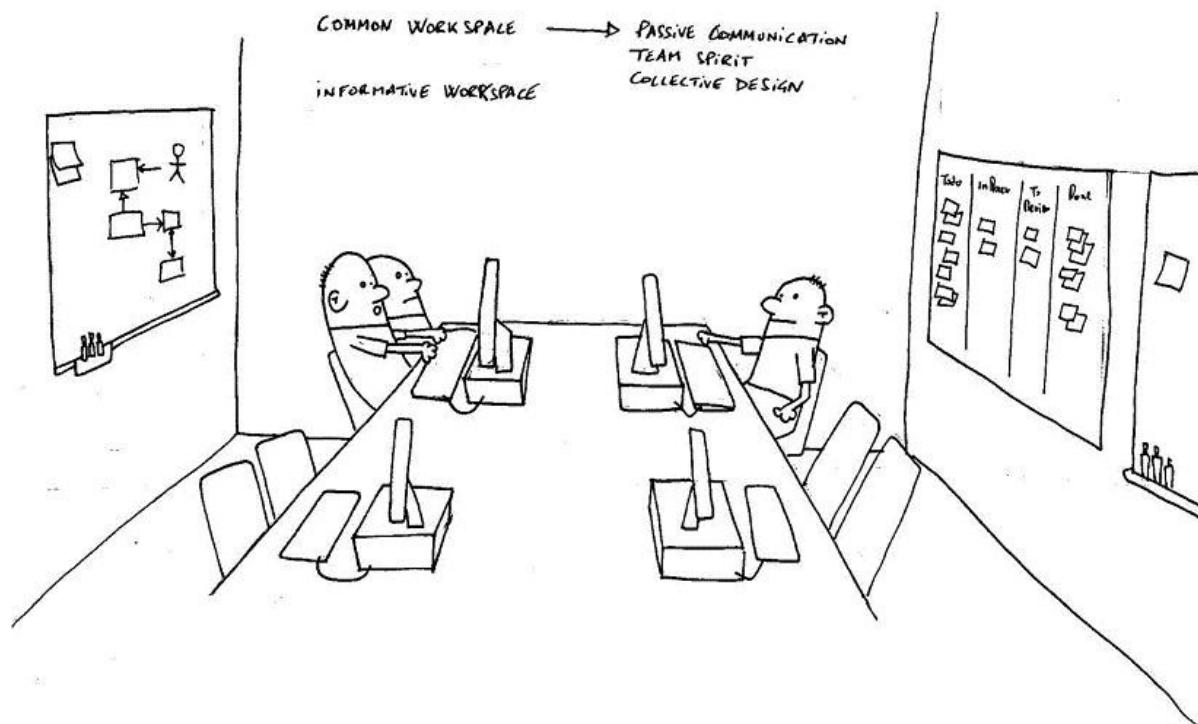
Praticiens du développement logiciel agile, nous avons trouvé dans l'eXtreme-Programming des recommandations pragmatiques pour organiser bureaux et postes de travail afin qu'ils favorisent le travail en équipe.

L'espace projet

Notre équipe de quatre développeurs logiciel partage un même bureau. Il est directement voisin de notre salle de réunion et du bureau partagé par le chef de projet et les experts métier. Cette proximité nous permet de privilégier la communication orale de face à face et renforce la cohésion de l'équipe.

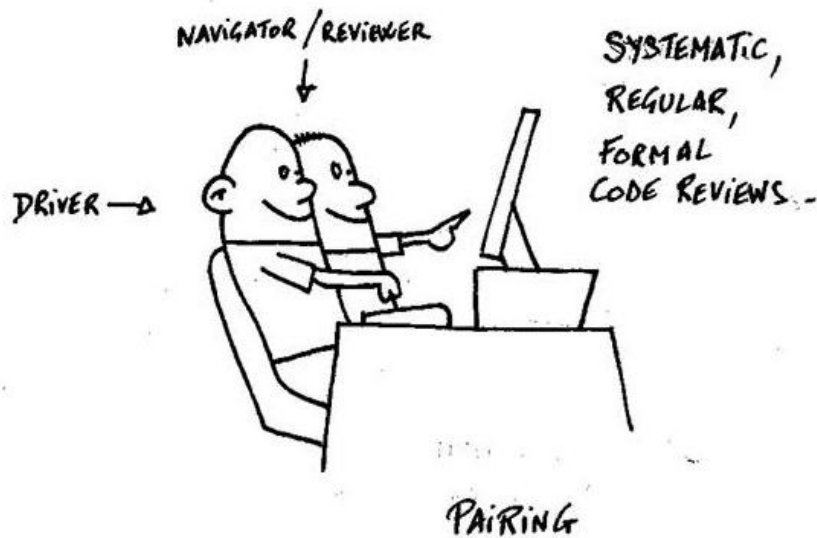
Le bureau du logiciel

Les développeurs logiciel sont tous installés autour d'une grande table afin que personne ne se tourne le dos. Nous communiquons toujours de face à face, sans avoir à interrompre les activités en cours. Ainsi tout membre de l'équipe peut aisément solliciter ses collègues pour le moindre conseil.



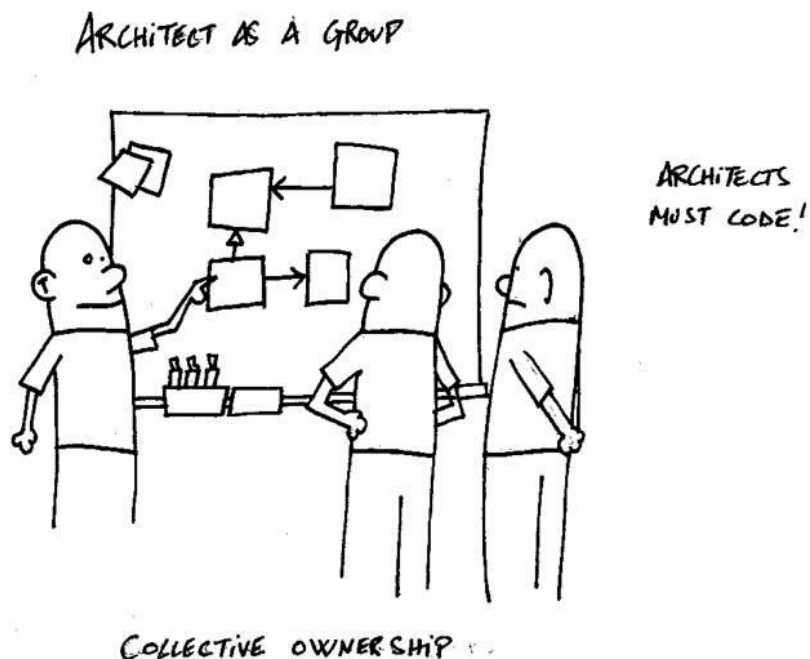
De plus, partager le même bureau favorise la communication passive: chacun est à tout moment au courant des activités des autres, même s'il n'est pas directement impliqué.

Dans cet espace, chaque poste est adapté au travail en binôme et aux relectures de code. En effet, chaque table peut accueillir deux développeurs qui s'échangent le clavier et la souris devant le même écran.

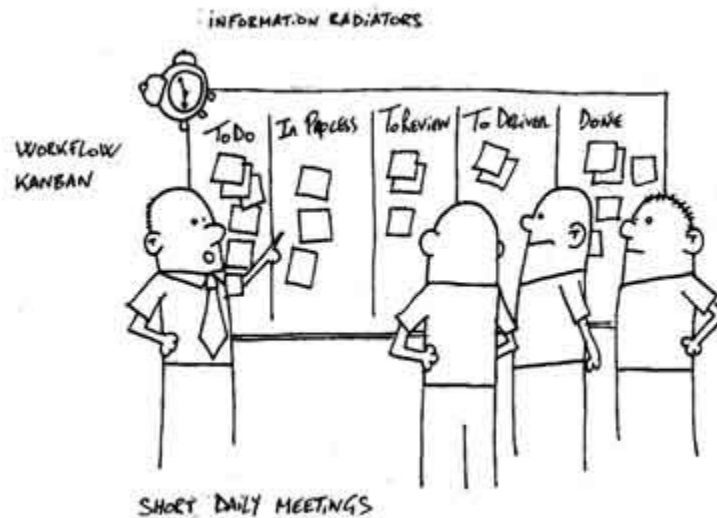


Tableaux blancs et post-it géants

Nous modélisons beaucoup en groupe et au marqueur en utilisant la notation UML. Les conceptions sur lesquelles nous sommes en train de travailler sont ainsi affichées en grand format. Ceci nous permet de revenir fréquemment sur nos décisions de design pour les conforter ou les remanier au fur et à mesure des informations que nous collectons en codant et en testant.



Un des tableaux est un grand kanban représentant notre flux de travail. Lors de réunions quotidiennes de moins de dix minutes, nous faisons avancer les tâches, représentées par de petits post-it répartis le long des cases du kanban. Cela nous permet de ressentir quotidiennement l'avancement de l'équipe et de l'afficher en toute transparence.



Un autre tableau est dédié aux améliorations potentielles. Chaque idée de remaniement de code, de test ou d'automatisation de procédure manuelle est résumée sur un post-it dédié collé sur le tableau. Les idées applicables peuvent être élues pour traverser le kanban du flux de travail.

Des post-it représentant les jalons du projet sont collés sur un grand calendrier affiché au mur. Ainsi, l'équipe ressent en un regard le temps qui les sépare de ces moments clés.

Au centre du bureau commun, un panier à courrier contient les numéros de [Pro]grammez ;o) En effet, nous discutons volontiers des articles qui nous touchent (intégration continue avec CruiseControl, tests de recette avec Fitnesse, développement model-driven avec génération de code contre des pratiques centrées sur le code et les tests, etc ...).

Ah, oui, nous avons un cinquième développeur dans le bureau: le poste d'intégration continue sur lequel tourne CruiseControl ;o)

Un environnement virtuel adapté au travail en équipe

Les développeurs partagent un outil de gestion de configuration, un outil de gestion des faits techniques, un Wiki pour échanger de l'information et un outil d'intégration continue qui publie la santé du logiciel et en notifie l'équipe par mail.

Bien sur, les développeurs utilisent un IDE, connecté à un compilateur natif et un compilateur croisé pour le processeur cible. Nous testons avec la suite xUnit, un outil de couverture structurelle et éventuellement Fit ou Fitnesse pour les tests de recette. Enfin, la panoplie est complétée par des outils de qualimétrie et de traçabilité des exigences.