



ADELI

La LETTRE n° 51

Avril 2003



Square des Utilisateurs

GCL : « l'initiative vient souvent de la Qualité »

Interview d'un Directeur Qualité dans un établissement financier

Quels sont les besoins réels à l'origine du déploiement de la gestion de configuration logicielle ? Qui sont les demandeurs ? La GCL répond-elle ou non aux exigences de l'entreprise et de son SI ? Quelles sont ses limites et ses frontières actuelles ? Quels sont les freins à son utilisation ? Quelles évolutions sont nécessaires et possibles ? Tels sont quelques uns des sujets abordés au cours d'un entretien avec le Directeur Qualité d'un important établissement financier, qui a accepté de dresser avec nous ce retour d'expérience.

Besoins et demandeurs

La gestion de configuration logicielle concerne plusieurs types d'acteurs informatiques mais, au fond, qui prend la décision de recourir à la GCL dans l'entreprise ?

On peut distinguer deux cas principaux de figure. Dans le cas d'une entreprise utilisatrice, c'est le plus souvent le DSI qui prend la décision. Dans le cas d'une société qui fabrique du logiciel, c'est soit la direction technique, soit la direction générale. En général, c'est donc la maîtrise d'œuvre : les gens du métier de l'informatique décident. Les utilisateurs, eux, ne se sentent pas les premiers concernés et ont tendance à penser que cela fait partie des « recettes de cuisine » des informaticiens.

Vous avez assumé différents rôles dans l'informatique et les entreprises ; quelles visions successives vous ont-ils apportés de la GCL ?

Effectivement, tout d'abord comme chef de projet, j'encadrais des équipes de développement à une époque où on ne parlait pas encore de GCL. Il y a trente ans, quand on développait des projets, on n'avait que des outils de gestion de bibliothèque et de gestion de versions, c'était tout. On arrivait tout de même à construire des logiciels fiables, sans outils sophistiqués, avec de la rigueur méthodologique.

La GCL est d'abord apparue dans le monde de l'informatique industrielle avec l'exigence du temps réel, tandis que pendant des années, en gestion, on faisait essentiellement du batch, de façon finalement assez simple, sans GCL véritable mais avec des gestionnaires de bibliothèques.

Par la suite, j'ai été responsable de production et j'avais en charge un service de time-sharing. Là aussi on faisait encore de la production sans GCL, mais on se débrouillait tout de même pour pouvoir assurer des retour arrière, avec les dispositifs appropriés.

On est venu à la GCL quand les systèmes sont devenus beaucoup plus complexes, dominés par le transactionnel et les mises à jour en temps réel des bases de données. Là, en cas de plantage, on avait beaucoup plus qu'avant besoin d'assurer ses arrières.

La GCL répond à un besoin de sécurité et de qualité

À quels besoins la gestion de configuration logicielle répond-elle ?

Avant tout à un besoin de fiabilité et de sécurité. À partir du moment où on a commencé à mettre des données à jour en temps réel, on s'est vite aperçu qu'on avait de gros pépins si l'on faisait des mises en production d'applicatifs un peu sauvages, sans possibilité de retour arrière.

Notez qu'il arrive encore qu'on ait de mauvaises surprises, même avec des outils de GCL et des gens qui pensent savoir les utiliser, parce que tous les objets réellement critiques ne sont pas considérés comme tels et ne sont en conséquence pas gérés en configuration. En particulier, des données, notamment de paramétrage de traitements, échappent souvent à la configuration, et alors des changements, de date ou autre, effectués à la volée, peuvent produire des catastrophes. On déploie des méthodologies complexes mais il y a encore des failles.

Pour le Directeur de la qualité, quels sont les rapports entre qualité et GCL ?

L'initiative de l'introduction de la GCL vient souvent de la qualité, en raison de la nécessaire application des normes informatiques. En effet, dans les règles régissant la relation client fournisseur, - par exemple la norme Z67 130, il y a toujours un paragraphe qui concerne la gestion de configuration. J'ai donc découvert la gestion de configuration par la qualité. Nous avions un ingénieur qualité venant de l'aéronautique qui avait ce souci là.

Par contre, pour les équipes de développeurs en informatique de gestion, c'était un peu comme du chinois, les textes normatifs semblaient ésotériques. Pour traduire et faire passer la GCL dans le monde de la gestion, nous nous sommes donc appuyés sur deux points précis : d'abord sur la gestion des environnements avec les règles de passage et de livraison d'un environnement à l'autre dans le cycle de développement, ensuite sur la gestion des versions. Nous avons commencé la diffusion uniquement avec un discours sur ces deux aspects. C'était déjà en soi un gros progrès, même si cette vision était assez restrictive.

Avec les systèmes répartis et l'arrivée d'Unix, des outils comme PCVS sont alors venus, mais on se trouvait malgré tout souvent avec des systèmes de gestion de configuration « à deux vitesses », suivant qu'on était sur les gros systèmes ou sur les systèmes répartis.

La faiblesse, c'est de ne pas tout gérer en configuration

Quelles sont, au travers de votre expérience, les limites de la GCL ?

Personnellement, j'exerce essentiellement dans un environnement qui produit du service sous forme de traitement de transactions financières. Le besoin de gestion de configuration est donc interne : c'est un souci de qualité pour la pérennité du business. Il y a, il est vrai, quelques environnements qui ne sont pas en gestion de configuration, des systèmes un peu spécifiques à la limite de la production et de l'informatique, pour le traitement des chèques ou des centres d'appels. Certains composants, comme la gestion de files d'attente sur un PABX ne sont ni entièrement matériels ni entièrement logiciels.

Je pense donc que l'on a besoin de sortir le «L» de la gestion de configuration « logicielle », celui-ci est un peu étroit. Le risque me semble aujourd'hui moins élevé dans le logiciel que dans le matériel et dans les données.

Par ailleurs, la frontière entre données et traitements me semble une frontière assez arbitraire. Par exemple dans la gestion d'un site Web, une page HTML, est-elle un programme ou une donnée ? La question a-t-elle un sens ? Est-ce que XML est de la donnée ou du traitement ? Le distinguo entre les deux, me semble un peu dépassé. C'est donc tout à fait logiquement, que la GCL est appelée à prendre aussi en charge « la gestion des contenus ».

La rétro-documentation, plus fidèle à la réalité de production

La documentation doit-elle être mise en configuration ?

On la met, plus ou moins. Mais les spécifications générales d'un système ne sont que rarement à jour. Si quelqu'un va modifier du code, je lui donnerais donc comme consigne de documenter d'abord la modification dans le source lui-même, avant d'aller mettre ou modifier un document Word quelque part. Je ne crois pas plus que cela à la documentation. La meilleure représentation d'un objet est pour moi l'objet lui-même, toute image n'étant qu'une image. On est toujours en droit, en effet, de se demander si la documentation est une représentation fiable, alors qu'elle l'est assez rarement.

Par contre, une documentation générée à partir du code est bien meilleure ; la rétro-documentation permet d'être plus sûr de la fidélité et de la cohérence.

Quels sont, finalement, les atouts principaux de la GCL ?

Les atouts, c'est d'abord la fiabilité du système, la possibilité de retour arrière à moindre frais. C'est donc un apport de sécurité en production. C'est en quelque sorte un petit mal pour en éviter un beaucoup plus grand. On a besoin de passer en configuration, quand il commence à y avoir besoin de beaucoup de communication entre les membres de l'équipe du projet, pour passer la responsabilité d'un intervenant à l'autre. Par exemple, après avoir passé ses tests unitaires, un développeur donne son programme à l'équipe d'intégration pour les tests d'intégration... On change alors d'environnement et on a besoin de savoir quel objet on manipule. Le risque commence là, mais il ne commence réellement que quand le moteur tourne, au même titre qu'une voiture qui passerait de l'atelier sur la route. Finalement, ce qui me détermine en matière de gestion de configuration, c'est le management des risques. En production, l'impact d'un pépin, cela peut être des millions d'euros en jeu, alors qu'en développement ce ne sont souvent que quelques jours hommes... Il faut donc se poser les questions par rapport à l'analyse des risques et à l'analyse de la valeur.

Les principales difficultés sont culturelles

Quelles sont, selon vous, les principales difficultés à la mise en place d'une GCL ?

C'est avant tout une difficulté d'ordre culturel et de formation. Les développeurs, les ingénieurs, ne semblent le plus souvent pas formés à la GCL. Est-ce qu'on enseigne la gestion de configuration à l'université ? : Je ne sais pas mais je n'en ai guère l'impression : comme si la programmation était toute l'informatique.

Il n'est que trop rarement encore question de méthode, de rigueur, de maîtrise des risques dans les cours, or cela ne s'improvise pas. Il faut des gens qui utilisent bien les outils, il faut au moins une ou deux personnes qui soient expertes sur chacun des projets, or elles sont sûrement encore trop rares sur le marché et doivent donc être chères.

Quant aux contraintes... Quand il y a de la pression, au jour le jour, sur les délais, on fait facilement des impasses, et c'est alors un risque.

Je suis dans une entreprise qui gère des transactions financières, des paiements par cartes bancaires. Or si un commerçant, parce que son client veut acheter un bijou de valeur avec sa carte bancaire, essaie de passer une autorisation et que le système d'autorisation ne marche pas, vous devez imaginer que dans le quart d'heure, le bijoutier appelle le directeur de la banque, et nous en avons tout de suite des nouvelles. Le problème peut être dû à une mauvaise mise en production informatique. En pharmacie ou en aéronautique, il peut y avoir, en plus de cela, des vies humaines en jeu.

Décloisonner la gestion du changement

Quelles évolutions nécessaires voyez-vous à la GCL et à la gestion de contenu ?

Tout d'abord, il serait bien qu'il n'y ait plus de cloisonnement matériel/logiciel, en particulier pour les composants réseaux, éléments hybrides, que l'on ne sait pas toujours où ranger. C'est aussi la même chose avec les données. Il faudrait donc pouvoir gérer les systèmes de bout en bout, en gestion de configuration.

Peut-on imaginer que la gestion humaine puisse aussi en faire partie ?

Pas encore... Mais la gestion des compétences existe : si quelqu'un s'en va, il est peut-être au moins aussi important, sinon plus, qu'un composant logiciel... Ce problème est documenté en analyse du risque.

Les entreprises se posent-elles suffisamment le problème de savoir de qui et de quoi elles ont besoin pour fonctionner ?

Au delà du risque financier qu'elles maîtrisaient assez bien, de nouvelles réglementations s'appliquent à présent aux banques sur le management des risques opérationnels, et cela devrait favoriser la GCL.

Le facteur humain doit bien être pris en compte en analyse des risques. Par exemple, la démotivation est un risque, et en cas de grève, la GCL ne sert pas à grand-chose.

Le Knowledge Management va, lui, capitaliser le savoir-faire de l'entreprise et situer sa connaissance. Le Knowledge Management, c'est en somme la gestion de configuration des connaissances¹.

*Propos recueillis par Mireille Boris,
Journaliste indépendante, membre de la commission GCL d'Adeli
mireilleboris@noos.fr,
assistée de Pierre Fischhof,
Chef de projet et consultant informatique, membre de la commission GCL d'Adeli
pierre.fischhof@libertysurf.fr*

¹ NDLR : cette affirmation, forcément discutable, nous semble ouvrir des perspectives, à la fois du côté de la GCL et du knowledge management.



Square des Utilisateurs

Convivialité ou efficacité

Du bon usage des nouvelles techniques de communication

Les nouvelles techniques de communication amplifient les débits de nos échanges d'informations ; désormais, nous transmettons, de plus en plus vite, des messages de plus en plus volumineux.

Mais communique-t-on vraiment mieux ? Ne consomme-t-on pas trop d'un temps précieux qui pourrait être mieux utilisé autrement.

À l'inverse, une communication trop succincte ne tuerait-elle pas une convivialité nécessaire à la cohésion d'un groupe ?

Nous vous soumettons ce petit billet d'humeur destiné à réfléchir sur le bon usage des nouvelles techniques de communication.

Nous éprouvons le plus grand plaisir à converser avec notre entourage. D'une façon informelle, nous véhiculons certaines informations, nous partageons nos commentaires, nous épanchons nos états d'âme. Cette aspiration à l'expression entretient une convivialité qui est un facteur essentiel de la vie sociale.

Cette communication n'est pas gratuite ; elle consomme un temps important, tant pour l'émetteur qui s'exprime que chez les destinataires tenus d'engranger les éléments transmis, de les analyser, et, quelquefois, d'y répondre. Les plages de temps ainsi consacrées à la communication sont prélevées sur un capital incompressible.

La communication s'effectue au détriment d'autres tâches directement productives. Certes, la réalisation des travaux prioritaires de production exige un sous-ensemble d'informations, précises et concises, issues des communications ci-dessus évoquées. Mais, ces informations utiles sont trop souvent noyées dans un torrent d'informations floues et superflues.

Faut-il privilégier la convivialité ou l'efficacité ? Un compromis est-il possible ?

Petit historique de la communication

De la communication synchrone rapprochée ...

Dans le passé, pour communiquer, deux individus devaient se rencontrer physiquement pour échanger, au moyen de messages sensoriels :

- ils se rapprochaient et se faisaient des gestes ;
- ils émettaient des sons plus ou moins articulés ;
- ils manifestaient leur affection par des caresses ou leur désaccord par des horions ;
- ils répondaient à l'appel des sensations olfactives (l'odeur de la cuisson du sanglier) ;
- une échelle de valeurs (de la douceur du miel à l'acidité du vinaigre) était attachée aux nourritures offertes.

On retiendra que ces communications primaires exigeaient le respect des unités de temps et de lieu.

... à la communication distante asynchrone

Puis, l'homme a codifié le message par la parole, le dessin, l'écrit. Ce qui lui a permis d'utiliser d'autres supports pour la transmission de ses messages.

L'homme a appris à s'affranchir de la distance. La voix peut se transmettre à quelques dizaines de mètres. Des signaux de fumées sont perceptibles à plusieurs kilomètres. Le télégraphe optique permet

de régénérer le message, tout au long de son trajet. Le téléphone permet des transmissions qui se jouent des distances.

Parallèlement, l'homme s'est affranchi de la simultanéité.

La poste transporte des plis écrits dont le contenu conserve une valeur dans le temps. Le télégramme et le fax accélèrent la transmission de ces documents.

Arrêtons-nous sur les récents apports des nouvelles techniques qui supportent nos communications distantes asynchrones.

Le téléphone mobile

Le téléphone mobile permet à deux individus qui se connaissent grâce à des identifiants numériques, de communiquer, quels que soient leurs emplacements respectifs. La conversation ne commence plus par le traditionnel « Allô » mais par l'onomatopée « Téou ? » (où es-tu ?) qui s'enquiert de la position géographique de l'autre interlocuteur.

Mais le téléphone mobile appelle la simultanéité. Car, bien peu de « mobilistes » utilisent la messagerie. Ils éprouvent le besoin de joindre, en direct, leur interlocuteur.

Le quémandeur, qui appelle, choisit son heure et le sujet qu'il impose à son interlocuteur ; il s'autorise, généralement, à troubler l'organisation de la personne appelée en faisant irruption dans ses activités présentes, sans éprouver la moindre culpabilité. L'émetteur fait quelquefois preuve de savoir-vivre en s'assurant que son destinataire partage son intérêt pour cette communication et qu'il en accepte les modalités de correspondance.

La messagerie électronique

Dernier avatar de la communication, la messagerie électronique offre des caractéristiques séduisantes.

Des fonctionnalités souples

L'émetteur peut envoyer le même message, intégrant du texte, de l'image fixe, de l'image animée, du son, à une liste de destinataires dont il ne connaît que les adresses électroniques. Il peut s'assurer de la bonne transmission de son message en sollicitant un avis de réception.

Chaque destinataire peut prendre connaissance des messages qu'on lui a adressés, au lieu où il se trouve et à l'heure qu'il lui plaît. Il a la faculté de retransmettre ce message à d'autres destinataires en y ajoutant éventuellement un commentaire. Il peut répondre à l'émetteur et éventuellement aux co-destinataires. Contrairement aux messages oraux rapidement volatils, les messages émis et reçus sont autant de documents électroniques appelés à être conservés.

Mais rien n'est parfait !

Des effets pervers pour le destinataire

Après quelques années de pratique, nous reconnaissons les multiples avantages de cet incontournable vecteur de communication. Cependant, dans la situation de destinataire, chacun de nous se plaint des principaux inconvénients engendrés par un usage, souvent désinvolte, de nos correspondants.

Un volume croissant

Nos boîtes aux lettres traditionnelles, alimentées par La Poste, nous livrent en abondance :

- courriers familiaux ... en décroissance régulière ;
- courriers administratifs ... dont les inévitables factures ;
- revues auxquelles nous sommes abonnés ;
- sollicitations pour des dons aux œuvres ;
- publicités adressées nominativement ;
- publicités anonymes généreusement ventilées.

De même, dans nos boîtes aux lettres électroniques, s'empilent pêle-mêle plusieurs familles de messages :

- courriels personnalisés et nommément adressés, émanant de nos correspondants professionnels, amicaux et familiaux ;
- flux régulier des abonnements aux lettres d'informations (newsletters) que l'on classe consciencieusement pour lire ultérieurement ;
- « spam », que les Québécois appellent élégamment « pourriel 1 », ensemble des messages publicitaires non souhaités ;
- copies des messages, généreusement envoyés par nos correspondants institutionnels mais qui ne nous concernent pas directement ;
- messages mal adressés.

Ainsi, l'internaute se transforme en orpailleur pour extraire d'un flot limoneux, chaque jour plus volumineux, les quelques précieuses paillettes indispensables à son activité.

Une identification difficile

Par économie (en fait par une paresseuse négligence) l'émetteur se contente souvent de reprendre un message précédemment reçu du destinataire qu'il veut joindre (technique du palimpseste² partiel), ce qui lui évite de reproduire l'adresse et l'objet. Le destinataire est alors contraint d'ouvrir le message et de lire pour connaître la nature du contenu, le plus souvent totalement étranger à l'objet indiqué.

Un arroseur arrosé

L'avis de réception systématique impose au destinataire le renvoi d'un message automatique qui vient remplir la boîte aux lettres de l'émetteur, pris à son propre piège.

La course à la priorité

Trop de priorités tuent la priorité. Un courrier vraiment urgent n'émerge plus d'un lot où tous les messages affichent le même niveau d'urgence.

Quelques recommandations pour l'émetteur

Avant d'émettre tout message

Quoi

Que veux-je dire ?

Ce qui se conçoit bien s'énonce clairement. Je préparer mon message en quelques phrases courtes, simples et claires. Je dois être capable de résumer ce message en un court libellé dans la fenêtre « objet ».

J'investis un peu de mon temps pour clarifier mes idées, pour les replacer dans le contexte de cet échange, pour les exprimer de façon synthétique, au plus grand bénéfice de mes destinataires.

Ceci m'amène à faire un message par objet. J'augmente le nombre de messages, mais mes messages, plus courts et plus ciblés, sont plus faciles à gérer par mes destinataires.

1 Typologie du « pourriel » - La LETTRE n° 34 de janvier 1999

2 Manuscrit sur parchemin dont on a fait disparaître l'écriture initiale avant d'y écrire de nouveau. Voir, dans ce numéro, l'article « Histoire de rasoir » et l'allusion au « nom de la rose ».

À qui

Destinataires impliqués

Le destinataire est l'interlocuteur privilégié que j'interpelle nommément. Le destinataire peut être un individu unique ou un groupe fonctionnel d'individus solidaires. Un groupe fonctionnel réunit plusieurs destinataires qui partagent le même intérêt vis-à-vis du message.

J'indique clairement pour quelle raison je sollicite mon correspondant : information, avis consultatif, et, éventuellement, souhait d'une réponse rapide.

Destinataires informés

Dois-je informer d'autres destinataires en leur adressant officiellement une copie du message ? Dois-je leur transmettre les pièces jointes ?

Destinataires officiellement informés

En général, les destinataires informés en copie pesteront à la réception d'un message qui les perturbe, lorsqu'ils en prennent connaissance à un moment inopportun.

En revanche, certains parmi ceux qui ne recevront pas copie de ce message se sentiront frustrés vous en tiendront rigueur.

C'est à l'émetteur de faire preuve de psychologie :

- en informant ceux qui seraient susceptibles de se manifester en cas d'omission ;
- et en épargnant ceux qui se plaindraient d'une abondance d'informations non souhaitées.

Destinataires discrètement informés

Faut-il que j'informe d'autres destinataires d'une façon confidentielle sans que les autres destinataires le sachent (copie aveugle) ?

Avant de clore la liste des différents niveaux de destinataires, il convient de vérifier l'exactitude et l'actualité des adresses électroniques très vulnérables à la bougeotte des destinataires.

En envoyant le message

Vérification du style

Le message est un document écrit qui pourrait être transmis facilement à des personnes externes à ma liste de diffusion. Aussi, est-il prudent de relire le texte pour vérifier les aspects suivants.

- Il est compréhensible des destinataires et ne comporte aucun risque majeur de mauvaise interprétation (involontaire ou volontaire).
- Il ne contient aucune remarque inutilement désobligeante, susceptible d'être exploitée à l'encontre des intérêts que je représente.

Liste des destinataires

Est-il pertinent de communiquer à tous les destinataires la liste des adresses des autres destinataires ? Pas de problème pour les groupes constitués. Mais, il n'en est pas de même pour certaines diffusions où l'un des destinataires pourrait être tenté d'utiliser ces adresses à d'autres fins, pour l'envoi de « pourriel » par exemple.

Il suffit de s'envoyer le message à soi-même et de mettre les adresses des destinataires en copie confidentielle.

Priorité

Pour être efficace, cette fonction doit être utilisée de façon exceptionnelle pour appeler l'attention du destinataire.

Avis de lecture

Quand dois-je demander un avis de lecture ?

- Lorsque je veux m'assurer qu'un message important et urgent a bien touché, en temps utile, le destinataire.
- Lorsque je n'ai pas confiance dans la rigueur de mon destinataire (il en est de négligents et de retors).

Après envoi du message

Lors d'un passé encore vivant dans les souvenirs de quelques-uns d'entre nous, pour chaque service une secrétaire professionnelle centralisait l'ensemble des envois de courriers. Cette personne imposait une certaine normalisation.

Cet intermédiaire n'existe plus. Il appartient à l'émetteur de gérer les documents qu'il crée : identification, mise en forme, ventilation des copies sur papier, conservation des archives.

Le souhait de respecter toutes ces recommandations devrait conduire les émetteurs à réduire sensiblement le nombre d'envoi en consacrant leurs efforts sur l'essentiel.

Qui s'en plaindrait !

Quelques recommandations pour le destinataire

À la réception du lot quotidien de messages, en tant que destinataire, je dois effectuer les tâches suivantes.

- J'élimine impitoyablement tout ce que je considère comme un pourriel : par utilisation d'un filtre et par destruction physique.
- Je signale à l'émetteur l'inutilité de me transmettre copie de messages relatifs à des thèmes qui ne me concernent pas directement ... actuellement.
- Je traite, immédiatement, les demandes personnalisées qui sollicitent une réponse brève et précise ; dans ce cas j'utilise la fonction « répondre » :
 - je veille à la pertinence de l'intitulé de l'objet ;
 - je n'envoie pas systématiquement la réponse aux co-destinataires ;
 - je ne renvoie que le dernier message auquel je réponds en élaguant les messages précédents. Sinon, chaque échange accroît l'historique d'une partie de ping-pong.
- Je diffère les réponses qui appellent une analyse plus longue (réflexion, recherche, concertation).
- Je classe les messages informatifs.

Pour toute information complémentaire, sachez qu'il existe un traité de savoir-vivre et Netiquette sur le site « www.ensmp.fr/Fr/Services/AutresServeurs/netqu.html ».

Conclusions

La sagesse consiste à user de tout sans abuser de rien. Sachons utiliser ces remarquables technologies pour améliorer notre efficacité. Évitions d'en abuser en créant artificiellement de l'information mal caractérisée.

Pour assurer de belles récoltes sur un terrain fertile, mieux vaut irriguer soigneusement que d'éclabousser à tous vents.

Alain Coulon
Secrétaire d'ADELI
info@adeli.org



Square des Utilisateurs

Estimation des charges d'un projet xNet

Méthodes d'estimation dans le cadre d'un projet xNet

D'après une thèse professionnelle réalisée à la DSI de la Caisse des Dépôts et Consignations (Établissement Public), dans le cadre de la formation du Mastère Spécialisé « Management des Systèmes d'Information et des Technologies », conduite conjointement par l'École des Hautes Études Commerciales (HEC) et l'École Nationale Supérieure des Mines de Paris (ENSM), sous la supervision d'Alain Berdugo (Professeur à HEC, Directeur du Mastère MSIT) et d'André Probst (Responsable Méthodes de la DSI CDC - Établissement Public)

Thèse consultable dans son intégralité sur :

<http://www.gezzed.net/these/These-MethodesEvaluationsChargesXnet.pdf>

Introduction

Lorsque l'on parle d'« Estimation des charges » de projets informatiques à une Maîtrise d'œuvre ou à une Maîtrise d'ouvrage, chacun a déjà plus ou moins une idée sur la question... sa propre méthode qu'il utilise, ou une vieille formule gribouillée sur une feuille (dont, d'ailleurs, on ne sait plus très bien d'où elle vient), mais qui a fait ses preuves, et que l'on continue à utiliser... ou alors, la bonne vieille méthode du « nez » du chef de projet, que l'on retrouve tout aussi fréquemment lors de l'estimation de charges de certains projets.

Évidemment, une entreprise qui a déjà fait un certain nombre de projets informatiques, connaît plus ou moins les charges et les délais dont elle aura besoin pour faire un projet, dont la plate-forme technique et les règles fonctionnelles lui sont familières. Ces « méthodes » d'une efficacité relative, n'en restent pas moins intuitives, et par là, perdent une certaine crédibilité lors d'une justification officielle de l'estimation.

Il existe pourtant des méthodes reconnues et efficaces d'estimation de charges de projet informatiques. Ces méthodes ont eu le temps de faire leur preuve dans divers domaines. La plupart d'entre elles se basent sur l'expérience d'un certain nombre de projets-types analysés.

Jusqu'à présent, ces méthodes étaient relativement fiables, vis-à-vis des projets informatiques « classiques », qui pouvaient se faire jusqu'au milieu des années 1990 – en l'occurrence, les applications monopostes, ou en client-serveur.

Mais depuis moins d'une dizaine d'années, et avec l'évolution exponentielle des technologies et l'apparition des xNet, les règles du jeu ont changé, une fois encore... Toutes les entreprises ont aujourd'hui leur internet, tous les grands comptes, leur intranet, et la majorité de commerces, leur extranet.

La mesure des charges de ces xNet est une « science » bien trop récente pour pouvoir en parler avec certitude, et sans risquer de se tromper. Mais, cette mesure devient aujourd'hui plus qu'indispensable, et l'utilisation d'une méthode sûre, et reconnue dans le cadre des xNet, pourrait aider à poser des bases concrètes de comparaison, pour les maîtres d'ouvrage et les maîtres d'œuvre et, pourquoi pas, les SSII.

L'évaluation de charges d'un projet Informatique

Pourquoi évaluer les charges ?

Dans le cadre des projets de la nouvelle économie, le concept de l'xNet est devenu quasi-incontournable : toutes les grandes entreprises possèdent aujourd'hui un intranet, et la conception (et le développement) de modules s'y greffant est le lot quotidien des responsables informatiques de ces entreprises.

Les diverses étapes de spécification, conception, recette, etc., prennent du temps, aussi bien du côté de la maîtrise d'œuvre que celui de la maîtrise d'ouvrage, et même de l'exploitation... et parfois beaucoup plus (ou beaucoup moins) que les métiers et la maîtrise d'ouvrage ne pourraient l'estimer.

Le concept de « Jour*Homme »

Le « Jour*Homme » est l'unité de mesure de la charge de travail dans le contexte d'un projet.

Concrètement, un projet estimé à « cent jours pour une personne à plein temps » pour arriver à son terme, on pourra considérer qu'il s'agit d'un projet estimé à « 100 jours hommes ». Cependant, nous pouvons nous permettre de supposer que tout projet, à un certain niveau, est « scindable » : c'est-à-dire que l'on va pouvoir le décomposer en un ensemble de tâches.

Si ce projet est partitionnable (et ce sera quasiment toujours le cas), il est raisonnablement possible d'effectuer plusieurs tâches en parallèle – donc de les faire effectuer par plusieurs hommes simultanément.

Partant de ce postulat, et dans un cadre idéal, on peut donc considérer qu'un projet de 36 J*H (planifié pour 12 hommes sur 3 jours) pourra être effectué en 9 jours par 4 hommes.

Cette répartition idéale suit une parabole. Nous verrons cependant un peu plus tard que cette théorie est loin d'être exacte dans le domaine qui nous intéresse, c'est-à-dire celui des projets informatiques de la nouvelle économie.

Les pièges de l'évaluation théorique

Il existe plusieurs pièges liés au fort désir des hommes à vouloir mettre au préalable une durée sur une tâche.

Par exemple, une des lois de C.N.Parkinson stipule que « les programmes sont comme les gaz parfaits : ils prennent toute la place qu'on leur donne ». L'interprétation de ce postulat est extrêmement simple : si l'on donne à une équipe un projet de 50 Jours*Homme (valeur « réelle ») à faire sur 100 Jours*Homme, le projet sera fait en 100 Jours*Homme, et pas un de moins. Cela fonctionne également dans l'autre sens : il est possible de réduire sensiblement la durée théorique d'un projet, et obtenir un résultat dans le délai escompté. Ce n'est cependant pas une méthode recommandée par la plupart des manuels de gestion de projet, et par les maîtres d'œuvre, eux-mêmes.

De plus, l'unité utilisée pour effectuer l'estimation de la durée est d'une fiabilité toute relative : reprenons l'exemple du projet de 100 J*H : selon la théorie, ce projet pourrait être effectué par 200 Hommes en une demi-journée. Bien que ce soit mathématiquement exact, c'est une aberration évidente : Les projets de l'informatique, en général, ne peuvent pas être scindés ainsi.

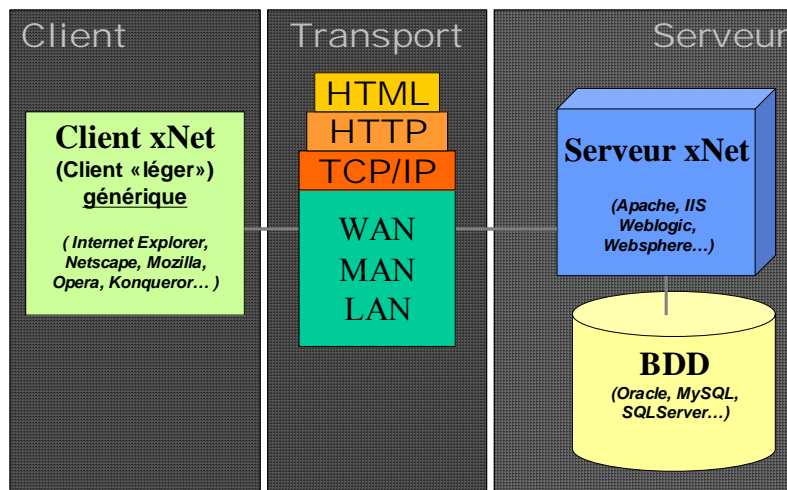
À l'extrême opposé, il existe également des projets qui ne sont pas partitionnables.

L'exemple revenant le plus souvent est celui –il est vrai, assez simpliste, mais très illustratif- de la femme enceinte, qui va mettre neuf mois à mettre au monde un enfant... Il est, de façon évidente impossible de partitionner, en plus d'une tâche. – Même trois femmes dotées de la meilleure volonté du monde ne pourront faire un enfant en trois mois.

Il faudra donc faire preuve de prudence lorsque l'on présentera une estimation de charge d'un projet, en tenant compte de tous les pièges tendus, amenés par le désir (bien compréhensible) des maîtres d'œuvre de mettre une durée sur une tâche ou un projet.

Le projet xNet

Le xNet



Le « xNet » (ou *Net) est une appellation générique du Système d'Information de front-office de l'entreprise. Il peut désigner un Internet, un Extranet, ou encore un Intranet, voire une combinaison de ces modèles de réseaux.

La différence fondamentale entre ces Internet, Extranet, et Intranet, est principalement liée au public (aux « visiteurs ») visé par ces réseaux, et par voie de conséquence au type d'accès de ces utilisateurs à ce que nous appellerons le « moteur de distribution de contenu ».

La plupart du temps, la technologie utilisée dans les xNet est la suivante :

Un serveur HTTP fournissant du contenu HTML, indifféremment statique ou dynamique, et interactif. À cela s'ajoutent divers services tournant derrière les serveurs HTTP, tels que des serveurs de données, ou de fichiers, et plus généralement quasiment n'importe quelle application pouvant potentiellement être assurée en back-office par le système d'information.

La frontière entre les trois concepts d'Intranet, d'Internet et d'Extranet s'avérant parfois extrêmement ténue, certains professionnels préfèrent utiliser le terme d'"Application xNet" pour désigner un système de « site » interactif mêlant indifféremment un ou plusieurs des trois concepts.

En général, on aura d'un côté, un ou plusieurs « clients légers » sur lesquels s'affichent une interface utilisateur, très proche du principe des interfaces standard (Windows ou MacOS), et dans lesquels le client verra s'afficher des objets graphiques lui permettant d'effectuer des opérations. Le résultat des opérations effectuées s'affichera de même dans cette fenêtre de « client léger" "

De l'autre côté, un ou plusieurs « serveurs », qui auront pour tâche de récupérer les ordres envoyés par le client, de les traiter, et de renvoyer au « client léger" le résultat des opérations.

- Un site dit « Internet » aura pour vocation de toucher toutes les personnes capables de s'y connecter physiquement.
- Un site « Intranet » est lié à un accès extrêmement restreint : le nombre de visiteurs potentiels est parfaitement connu, ainsi que l'identité de ces derniers. Ce type de site requiert en général des technologies dynamiques, et apporte des services conséquents à ses visiteurs, dans le cadre, par exemple, d'une organisation.

- Un site « Extranet », plus restreint qu'un site Internet, sera destiné à un public possédant un mot de passe, ou une clef permettant de consulter ses données. Le plus souvent, ce type de site est dynamique, et les visiteurs sont a priori connus, ou, en tout cas, répertoriés.

Caractéristiques intrinsèques

Grâce au modèle HTML et aux liens hypertextes (inventés par Théodore Nelson en 1965), on peut, de façon extrêmement simple, modéliser n'importe quel xNet sous forme d'un ensemble de « pages » liées entre elles par ces liens. On trouve par ailleurs souvent dans les xNet un module de navigation (intimement lié à un module d'identification), permettant au xNaute de naviguer suivant leur profil.

Suivant le type de xNet visité, et suivant le contexte dans lequel on se place, tous les utilisateurs n'auront pas accès de la même façon au site... (ces « droits d'accès » ayant une répercussion directe sur le module de navigation). Ceci est parfaitement compréhensible si l'on considère que derrière chaque page visitée, il y a une information (pouvant être à caractère privé), ou un traitement (pouvant être à caractère sensible) qui peut être uniquement dédié à un groupe d'utilisateurs, voire à un seul utilisateur.

Exemples : authentification d'un site Web d'une banque (Extranet), administration distante d'un site, etc.

Gardons en tête qu'un xNet n'est potentiellement pas un simple « site web ». C'est une application à part entière : en plus de l'interface utilisateur, chaque xNet possède son traitement et ses données.

La partie « visible » du xNet, l'interface, est en fait un contenu dynamique affiché sur un client... ce contenu est le résultat d'une suite d'opérations effectuées sur un ou plusieurs serveurs distants, grâce à des données situées elles aussi, sur un ou plusieurs serveurs distants. La seule différence significative par rapport au Client-Serveur, est que l'interface client est normalisée, et que les transactions liant l'interface aux traitements le sont aussi.

Il est donc tout à fait légitime, dans une certaine mesure, de considérer le xNet comme une application Client-Serveur, et – en terme de charges - de l'évaluer de façon similaire.

Les différentes méthodes que nous allons analyser ne prendront pas en compte le contenu textuel statique du site, qui devra faire l'objet d'une spécification à part, du côté de l'utilisateur.

État de l'art des méthodes d'évaluations

Méthode des points de Fonctions

La méthode des Points de Fonctions est destinée à évaluer la taille d'un projet indépendamment de la technologie utilisée pour le réaliser. Son avantage réside principalement dans le fait qu'il est possible d'effectuer une évaluation grâce à cette méthode très en amont.

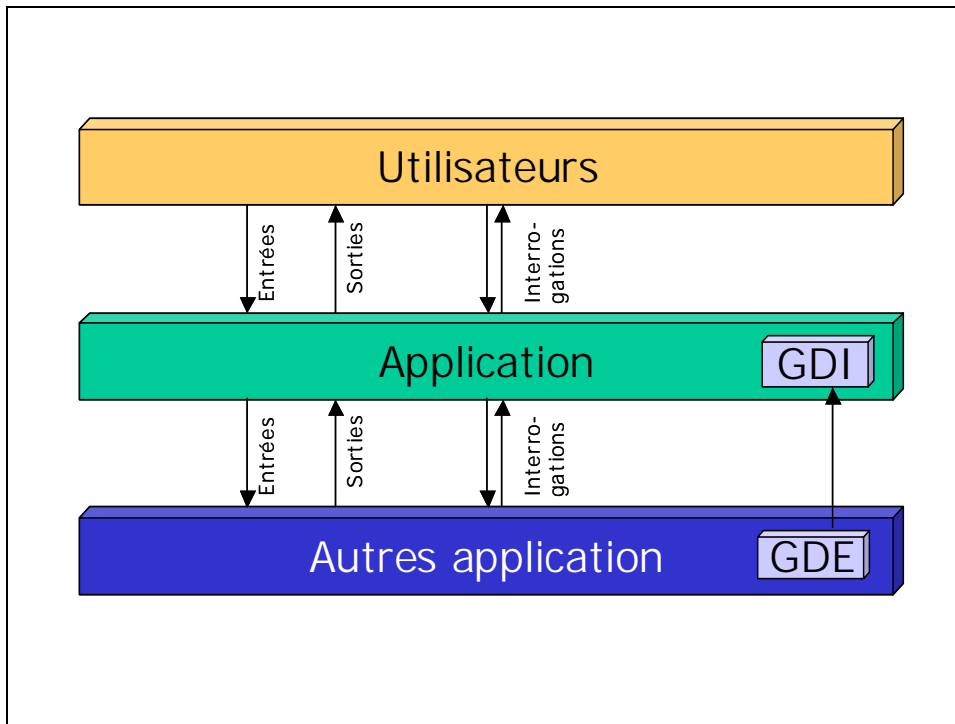
On s'affranchit, de plus, du comptage de lignes de code – comme pour la méthode COCOMO, qui devient d'une fiabilité tout à fait relative en fonction du langage utilisé... Ceci étant d'autant plus vrai dans le domaine des xNet ou une grosse partie des lignes de « code » définit l'interface.

Les points de fonctions reposaient, à l'origine, sur un calcul basé sur quatre entités (entrée, sortie, interrogation, fichiers), et ce, sans catégorisation de complexité. (Albrecht, 1979).

Depuis le milieu des années 80, avec l'IFPUG (International Function Points User Group), et la normalisation AFNOR, le comptage des points de fonctions se fait à partir des entités suivantes :

Données internes (GDI) : Groupe de données logiquement liées, ou de groupe de paramètres de contrôle, et identifiables par l'utilisateur. Ces données sont mises à jour et utilisés à l'intérieur de la frontière de l'application. Les entités reliées par une cardinalité (1,1) forment un seul et même GDI.

Données externes (GDE) : Groupe de données logiquement liées, ou groupe de paramètre de contrôle, et identifiables par l'utilisateur. Ces données sont utilisées par l'application, mais mises à jour par une autre application. Le GDE est un GDI dans un autre domaine.



Entrées (ENT) : Données, ou les paramètres de contrôle, qui entrent dans l'application considérée. Ces entrées maintiennent un ou plusieurs GDI, initialisent ou contrôlent un traitement, et font l'objet d'un traitement unique. Une Entrée correspond donc à un écran de saisie, ou à une réception de données. A chaque GDI doit correspondre au moins une entrée, permettant sa mise à jour.

Sorties (SOR) : Données, ou les paramètres de contrôle qui sortent de l'application. Ces sorties sont le résultat d'un traitement unique (différent d'une simple extraction de données). Ce peut être un écran de visualisation, ou un message vers une autre application.

Interrogations (INT) : Ce sont des données élémentaires qui correspondent à une extraction de données. L'INT ne met à jour aucun GDI.

On associe à ces cinq entités des paramètres supplémentaires, qui vont permettre de déterminer par la suite le niveau de complexité de chaque élément.

Données élémentaires (DE) : Chaque GDI ou GDE est composé de données élémentaires. Une DE équivaut à un champ de données. On compte un seul DE par champ répétitif dans les entrées, les sorties, et les interrogations.

Sous-ensemble logique de données (SLD) : Groupements logiques de GDI ou de GDE qui sont traités simultanément dans l'application.

Groupe de données référencées (GDR)

Groupements logiques de GDI ou de GDE qui sont mis à jour, ou consultés simultanément par les différentes ENT, SOR ou INT.

Points de fonctions				
		1 à 19 DE	20 à 50 DE	> 50 DE
GDI	1 SLD	7	7	10
	2 à 5 SLD	7	10	15
	>5	10	15	15
		1 à 19 DE	20 à 50 DE	> 50 DE
GDE	1 SLD	5	5	7
	2 à 5 SLD	5	7	10
	>5	7	10	10
		1 à 4 DE	5 à 15 DE	> 10 DE
ENT	0 ou 1 GDR	3	3	4
	2 GDR	3	4	6
	> 2 GDR	4	6	6
		1 à 5 DE	6 à 19 DE	> 19 DE
SOR	0 ou 1 GDR	4	4	5
	2 ou 3 GDR	4	5	7
	> 3 GDR	5	7	7
		1 à 5 DE	6 à 19 DE	> 19 DE
INT	0 ou 1 GDR	3	3	4
	2 ou 3 GDR	3	4	6
	> 3 GDR	4	6	6

Les points de fonctions sont calculés en fonction du tableau ci-contre, selon le nombre d'entités et de leur complexité.

L'étape finale consistera à ajuster ces points de fonctions grâce à un ensemble de 14 degrés d'influence. (Communication, distribution des données ou des traitements, performance, intensité d'utilisation de la configuration matérielle, taux de transition, de transaction, saisie interactive, convivialité, mise à jour en temps réel des GDI, complexité des traitements, réutilisation, facilité d'installation, facilité d'exploitation, portabilité, facilité d'adaptation).

Ce Facteur d'Ajustement va nous permettre de corriger le nombre de points de fonctions bruts avec un facteur allant de 65% à 135% (donc avec un résultat allant potentiellement du simple au double).

Par la suite, la charge se calcule grâce à un facteur multiplicatif, d'environ 3 Jours*Homme par Point de Fonction en moyenne (2 J*H si le projet est petit, 4 s'il s'agit d'un grand projet)

À la fin de l'étude détaillée, ce facteur peut varier de 0,1 (pour les projets en L4G, dans le meilleur des cas) jusqu'à 2 pour les projets plus complexes, en passant par un facteur 0,5 pour les projets de type RAD, où la productivité est plus élevée.

Pour synthétiser, le comptage subjectif, difficile à automatiser, et même si l'estimation de l'effort est juste, l'estimation de la charge (sujette à une multiplication par une valeur subjective) est d'une exactitude toute relative. Cependant, retenons que l'estimation est d'une part disponible assez tôt dans le projet, et surtout, est indépendante du langage, de la plate-forme, et des diverses technologies utilisées.

On peut, avec l'expérience et les statistiques, adapter facilement le modèle à l'estimation des xNet.

Méthode COCOMO (Constructive Cost Model)

« Pour estimer la charge d'un projet, il faut en estimer la taille ». La principale hypothèse de cette méthode est qu'un informaticien saura mieux évaluer la « taille » d'un projet informatique que l'« effort » à fournir pour le développer.

COCOMO a été défini par B.W.Boehm sous la forme d'équations simples, admettant quatre paramètres.

Ces divers coefficients ont été trouvés de façon expérimentale, Boehm se basant sur l'estimation et la réalisation de plus d'une soixantaine de projets divers d'informatique classique. (chaque projet ayant une taille située entre 2000 et 100.000 lignes de code) La probabilité d'obtenir un résultat proche de la réalité grâce à COCOMO dépend du fait que le projet que l'on analyse est proche ou non du panel de projets analysés par Boehm.

Le paramètre principal est la « ligne de code ». Le nombre de milliers de ligne de code représente la « taille » du projet. Selon N.E.Fenton (Software Metrics: A Rigorous and Practical Data, 1998) : « Une ligne de code est toute ligne du texte d'un programme qui n'est pas une ligne de commentaire, ou une ligne blanche, sans considération du nombre d'instructions ou de fragments d'instructions dans la ligne. Sont incluses toutes les lignes contenant des en-têtes de programmes, des déclarations, et des instructions exécutables et non exécutables. ».

Cependant, dans le contexte des xNet, contrairement aux langages modernes, orientés GUI, on est en droit de se demander si le concept de « nombre de lignes de code » reste fiable, en raison de la nécessité d'utilisation d'un grand nombre d'entre elles pour définir uniquement l'interface de l'application xNet. (C'est typiquement le cas du HTML).

La taille d'un projet est estimée en milliers de lignes de code. On parle généralement de KDSI (Kilo Delivered Source Instruction), ou de KLOC. (Kilo Lines Of Code)

Il est possible, théoriquement, d'évaluer la taille (le nombre de KDSI) en fonction du nombre de points de fonctions du projet, grâce à la formule $T = 0.1187.PFA-6.49$.

Une autre méthode d'estimation de T peut être déterminée grâce aux facteurs suivants, en fonction du langage utilisé : (Assembleur : 0,32 KDSI par Point de Fonction, C : 0,15 ; COBOL : 0,106 ; Pascal : 0,091 ; Prolog : 0,064 ; Basic : 0,064 ; SQL : 0,040 ; Smalltalk : 0,021)

La première chose à faire lorsque l'on décide d'utiliser COCOMO pour évaluer le projet, est donc de déterminer à quel « type » de projet l'on a affaire. Boehm propose trois types de projet :

Projet Organique : Projet simple, ou de routine, effectué par une équipe ayant déjà travaillé ensemble, dans lequel il y a peu de « surprises » et où une bonne anticipation est possible.

Projet Semi-Détaché : Entre organique et imbriqué. Le Projet n'est ni trop simple, ni trop complexe, l'équipe de développement se connaît un peu, et les technologies peuvent être mal connues, mais pas d'une grande difficulté d'appréhension.

Projet Imbriqué : Techniques innovantes, organisation complexe, beaucoup d'interactions. Projet difficile, ou dans un domaine inconnu par l'entreprise, équipe de développement n'ayant pas encore travaillé ensemble, ou projet impliquant des technologies encore peu connues des développeurs.

Les calculs se font grâce aux équations suivantes :

COCOMO Intermédiaire	a	b	c	d
Organique	3,2	1,05	2,5	0,38
Semi-détaché	3,0	1,12	2,5	0,35
Imbriqué	2,8	1,20	2,5	0,32

$$\text{Effort} = 21 * a * T^b * \text{Facteur d'Ajustement}$$

$$\text{Durée} = 21 * c * (\text{Effort}/21)^d$$

$$\text{Effectifs} = \text{Effort}/\text{Durée}$$

Le facteur d'ajustement total est égal au produit de 15 facteurs d'ajustement, en quatre catégories :

- Attributs du produit (Fiabilité Requête, Taille de la Base de Données, Complexité du produit)
- Attributs du Matériel (Contraintes de temps d'exécution, Contraintes de taille mémoire principale, Instabilité de la Machine Virtuelle, Temps de Retournement)
- Attributs de l'équipe (Compétence des Analystes, Expérience du domaine d'application, Compétence des Programmeurs, Expérience de la Machine Virtuelle, Expérience du langage)
- Méthodes et Outils (Pratique des Méthodes, Utilisation d'outils logiciels, Contraintes de planning)

Notons que le facteur d'ajustement peut varier de 0.09 à 72.38 (un rapport de plus de 800 !!!), sa valeur étant égale à 1 pour un projet normal.

Pour synthétiser : il est assez ardu de calculer de façon fiable la « taille » du projet (en KDSI) lorsque l'on est peu avancé dans le projet, et la fiabilité du résultat obtenu dépend énormément du fait que le projet à réaliser se rapproche fortement de ceux qui ont servi à créer le modèle.

Le fait d'utiliser la « ligne de code » en tant qu'unité de calcul amène quelques failles dans le système : on ne tient pas compte, par exemple, des commentaires dans le code (Qui, pour la maintenance ou le débogage s'avèrent primordiaux).

COCOMO présente toutefois l'avantage d'être transparent : il est facile d'ajuster ses paramètres et de recréer son propre modèle.

ÉVALUATEUR-RAD (J-P. Vickoff)

ÉVALUATEUR-RAD est un logiciel de Jean-Pierre Vickoff qui se base, selon l'auteur, sur une évolution des travaux de Albrecht, Ashby, Boehm, Clark, Egyed, Gacek, Hoh, Lehman, Martin, Maxwell, Pareto, Parkinson, Rayleigh et Shannon... autant dire une combinaison de la plupart des méthodes d'évaluation et de statistiques du domaines. Ce n'est donc pas, à proprement parler, une méthode. Cependant, ce logiciel (disponible sur <http://mapage.noos.fr/rad/eval2000.htm>) reste néanmoins très intéressant dans le contexte qui nous intéresse, dans la mesure où Évaluateur-RAD prend en compte plusieurs paramètres, dont la nature a totalement été occultée dans les autres méthodes analysées ici. De plus, il s'agit bel et bien du produit la plus « adapté » aux xNet, les autres méthodes répondant aux demandes d'évaluation, pour la plupart, du client-serveur, ou bien de l'informatique classique.

Après plusieurs tests approfondis du dit logiciel, plusieurs personnes estiment que la charge jaugée en fin de remplissage des écrans était légèrement supérieure à la réalisation réelle.

Nous vous conseillons d'essayer vous-même ce logiciel. Son utilisation est intuitive, et on obtient un résultat rapidement.

Cependant, il semble, au final, donner des chiffres trop élevés pour des petits projets. De plus, certains coefficients de pondération sont difficilement explicables ou justifiables (ex: taille de l'écran de l'application, etc...), qui amènent, par exemple, pour un projet nominal de 100 Jours*Homme, une variation de 8% à plus de 35 000% de la taille prévue.

Méthode Interne CDC (A.Probst)

Cette méthode a été mise au point par André Probst, Responsable Méthodes de la Direction des Systèmes d'Information de la Caisse des Dépôts et Consignations. La méthode correspond, à l'origine, à l'évaluation de projets de type client-serveur.

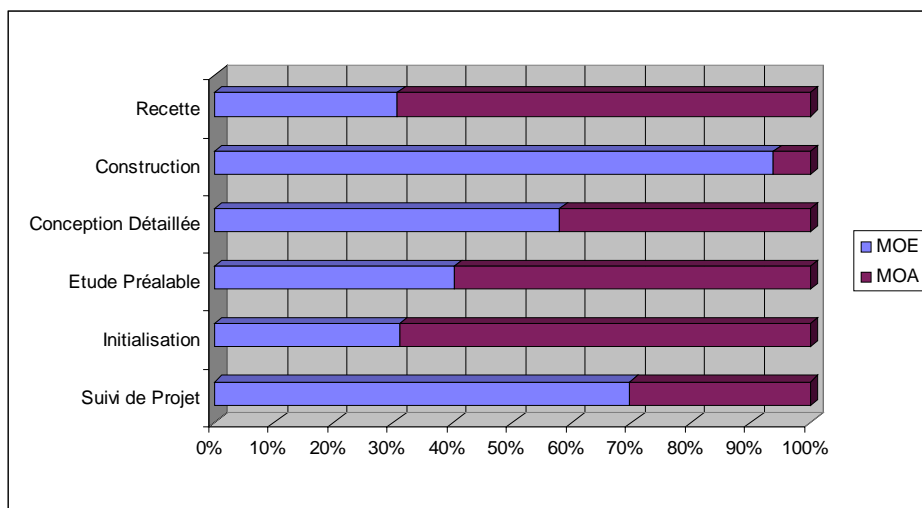
Les coefficients ont été obtenus grâce à l'expérience, sur des essais fait sur plus d'une dizaine de gros projets de la Caisse des Dépôts et Consignations. Cette méthode a l'avantage d'évaluer à la fois, la charge nécessaire en terme de maîtrise d'œuvre et d'ouvrage, en rentrant dans tous les détails de la construction de l'application.

La première étape consiste à décomposer le projet en macro-fonctions. Chacune de ces macro-fonctions correspond à une fonctionnalité globale de l'application. Par la suite, il s'agira de décomposer chaque macro-fonction en un ensemble de micro-fonctions, ces dernières étant pondérées par leur niveau de complexité que leur attribue le chef de projet grâce à sa propre expérience.

Puis il convient de calculer un « Nombre d'Objets/Relations ». Sa valeur correspond, dans le MCD, à la somme du nombre de tables « significatives » du modèle de données, et du nombre de relations les liant.

On obtient donc dans un premier temps un « Effort » (il s'agit de la métrique de référence) à fournir dans le cadre de la réalisation d'un projet, à partir d'une étape avancée dans le temps, correspondant approximativement au milieu de l'étude préalable. (La méthode d'origine permettant de calculer les charges en fonction de ces informations était basée sur une durée de projet, évaluée à partir de cette période).

Cet effort est décomposé grâce à un certain nombre de coefficients en un certain nombre de charges.



La répartition des charges du projet se fait pour la maîtrise d'œuvre et la maîtrise d'ouvrage en fonction de diverses pondérations de la charge, suivant des coefficients affectés à certaines tâches du projet.

Il faut garder en tête que les étapes et les coefficients calculés ici sont adaptés au Client-Serveur ; pour pouvoir être en adéquation avec les xNet, A.Probst propose d'éventuellement diminuer la charge du projet de 20% par rapport à sa valeur d'origine (pour avoir un résultat rapide), ou mieux, ajouter des étapes, et adapter les différents coefficients.

Par la suite, certaines charges calculées précédemment sont directement liées au niveau d'expertise de l'équipe de développement. On va ainsi pouvoir les pondérer. On compte parmi ces tâches le Maquettage, l'Architecture Applicative, les Spécifications Techniques, le Codage des Composants, et les tests.

On applique à chacune de ces charges un coefficient de pondération en fonction du niveau d'expertise du développeur.

Puis, on va estimer un besoin « transverse » : un surcoût lié à des structures transverses, s'impliquant principalement dans le projet lors de réunions, pour des besoins de suivi de projet en terme d'architecture, de qualité, et de sécurité.

On pourra ainsi calculer la charge du projet en extrayant les étapes correspondant à la conception, la construction du projet, et sa recette.

Méthode interne ICDC (ESTIMAT)

ESTIMAT est un outil mis au point au début des années 1990 par une cellule d'Informatique CDC.

Cette méthode se base principalement sur la méthode Algorithmique : le principe consiste à utiliser des formules mathématiques pour exprimer l'effort en fonction d'éléments mesurés, ou estimés.

Par la suite, on leur applique un coefficient d'ajustement se basant sur l'environnement. (Le principe de ces coefficients est relativement similaire au modèle COCOMO). Ce sont des données issues de l'expérience qui vont servir à déterminer ces coefficients.

Le but de la première opération est de calculer un effort tenant compte, à la fois de la complexité des règles de gestion et des règles de mise à jour de la base, mais aussi du volume des travaux à effectuer.

On distingue ainsi deux types de traitements :

- Le traitement Conversationnel : (ou Transactionnel) - Nombre d'écrans dédiés à l'utilisateur
- Le traitement par lots : Nombres d'états – Traitement automatisé

<i>ESTIMAT</i>				
<i>Estimation détaillée</i>				
JH		Simple	Moyen	Difficile
Traitements conversationnels	Fixe	5	10	15
	Variable	3	4	5
Traitements par lots	Fixe	4	8	12
	Variable	4	6	8

À l'intérieur de ces traitements, on va distinguer deux parties :

- Partie Variable : écrans, ou traitements.
- Partie Fixe : Traitement de la partie variable, Processus. (Coût de la séquence entre les parties variables)

L'effort est calculé en sommant les valeurs associées aux types et à la complexité des divers traitements suivant le tableau ci-dessus. On pondèrera cet effort grâce à un coefficient résultant d'ensemble de facteurs, similaires à la méthode COCOMO. Ce coefficient peut varier de 0.12 à 7.5 (un rapport de plus de 60... ce qui reste très raisonnable par rapport à COCOMO, mais reste extrêmement élevé dans l'absolu), sa valeur étant égale à 1 pour un projet normal.

Cette méthode ESTIMAT est visiblement inspirée d'une métrique de type « Points de Fonction » pour l'estimation nominale de la charge, et de COCOMO pour la pondération. Elle a l'avantage indéniable d'affecter à la charge nominale des coefficients permettant d'estimer la charge totale, des spécifications à la mise en production. La méthode est, a priori, bien adaptée à tous les projets en général, puisque ne dispose pas de paramètres spécifiques au type d'application développée. Il est donc aisé, une fois de plus, d'adapter les coefficients de la méthode pour obtenir une estimation propre aux xNet.

Évaluation d'un xNet

Vue d'ensemble

Les méthodes que nous avons analysées jusqu'à présent (hormis celle de J-P.Vickoff) ne sont pas, à proprement parler, dédiées à l'évaluation des xNet. C'est d'autant plus cohérent, que ces méthodes datent d'avant le milieu des années 1990.

Il est relativement légitime de considérer le développement d'un xNet comme celui d'une application en client-serveur (et ce, pour des raisons techniques). Cependant, on notera tout de même quelques différences flagrantes entre ces deux mondes, qui ne sont pas sans avoir un impact important sur un paramètre de développement qui nous intéresse particulièrement ici, à savoir la charge nécessaire au développement.

Tout d'abord, il faut savoir que le client sur le xNet est universel (c'est le navigateur). Le code de l'application « client » s'en trouve de fait considérablement simplifié : grâce à un « méta-langage » à la fois simple, et ayant un formidable potentiel en terme d'interface, il n'est plus nécessaire d'avoir un code lourd du côté client... sans compter que la simplicité du HTML permet une capacité de modification du code client sans comparaison. De même, grâce à la standardisation du CGI et du HTTP, on s'affranchit des protocoles complexes de communication (Sockets, RMI, RPC, CORBA, SOAP, COM, etc.) inhérents au client-serveur.

Ensuite, nous avons pu voir dans certains méthodes que les applications de type « RAD » pouvaient être développées beaucoup plus rapidement... pour dire la même chose d'une autre façon, si l'utilisateur peut donner le plus souvent possible un retour sur l'application en cours de développement, les chances de voir le projet dévier, seront amoindries (dans une certaine mesure, bien entendu), et le développement devrait en être d'autant plus rapide.

Dans la réalité, ce genre de fonctionnement n'est pas forcément du goût de la maîtrise d'œuvre... en effet, si ce levier est mal utilisé par la maîtrise d'ouvrage, la pression subie par les développeurs, et les changements d'opinions (inévitables) de la part des maîtrises d'ouvrage pourrait également faire perdre du temps de développement.

Il faut aussi bien voir que si une partie de la charge de la maîtrise d'œuvre est réduite, du fait d'une vérification assez assidue de la maîtrise d'ouvrage, ce gain de charge est contrebalancé de l'autre côté par une trop grande implication et une perte de temps pour la maîtrise d'ouvrage... ce qui est également coûteux.

Enfin, il existe divers problèmes de « sécurité » dus au fait que le client soit standardisé... ceci pouvant nécessiter une charge supplémentaire, assignée au contrôle du déroulement correct du processus, aux vérifications des authentifications, au comblement des trous de sécurité, etc.

En bref, le type d'application xNet peut aussi bien se trouver être largement plus simplifiée en terme de développement que le client-serveur, ou bien nécessiter à peu près la même charge, voire complètement dépasser les délais prévus, et ce, en fonction principalement, de la complexité de l'application... et des contraintes annexes.

Modules xNet

Il faut savoir, par la suite, que le développement d'un xNet connaît plusieurs étapes potentielles. La première, est à proprement parler, celle de la « création », et la seconde, celle de l'« ajout de modules ».

La création d'un xNet est l'étape qui a lieu avant toutes les autres. (les autres étant des « ajouts de modules », si le cas se présente).

Il s'agit en fait, de la toute première réalisation, de la « création » à proprement parler. Lors de cette étape, sont établies toutes les bases de l'application : framework applicatif, module d'authentification, module de navigation, charte graphique, etc. et, bien évidemment, les modules métiers, pour lesquels le xNet a été mis en place.

La seconde étape pourrait être comparée à une simple « maintenance évolutive » : il suffit de greffer à l'existant une application possédant ses propres règles fonctionnelles. Le terme « greffer » est tout à fait adapté : il s'agit en effet de se baser sur les modules d'identification et de navigation existants (et au besoin, les modifier) pour ajouter une application à l'existant.

Ce type de « projet » est beaucoup moins spécifique que lors d'une création, et s'apparente, dans ce cas, plus à de l'informatique classique.

Il faut savoir que les solutions techniques de créations de xNet sont multiples. Celles-ci peuvent aller de la simple page en HTML sans base de données, à des technologies objets métiers (Beans, EJB,

Java...), des procédures stockées en base (PL/SQL...), en passant par des langages pré-compilés (CGI en général...), ou interprétés côté serveur (Perl, ASP, PHP...).

Bref, la maîtrise d'œuvre dispose d'un panel de choix technologique on ne peut plus important. Le choix en est d'autant plus difficile qu'un même résultat peut être obtenu en utilisant des technologies tout à fait différentes, parfois de la plus simple à la plus complexe.

En fait, tout dépend du niveau de réutilisabilité de l'application, du besoin en maintenance, de la performance désirée. Il va falloir, quels que soient les délais imposés, trouver le meilleur compromis entre le temps de développement, les contraintes de plate-forme technologique, le besoin en réutilisabilité et en maintenance, etc.

D'autant qu'il est de plus en plus nécessaire de revenir par la suite sur les applications, pour des besoins en terme d'ergonomie, ou d'évolutivité du produit. Le temps passé en développement doit être considéré comme un « investissement » sur les maintenances à venir. (Comme dans tous les projets informatiques, d'ailleurs).

Alors comment évaluer le xNet ? La méthode idéale pour évaluer un xNet serait, de façon intuitive, d'adapter un modèle existant (La méthode Probst, par exemple, étant la plus transparente, est très adaptée) en modifiant les coefficients attribués.

Un exemple concret

Le projet RPI

Le projet RPI est un projet interne de la DSI de la Caisse des Dépôts et Consignations, qui a pour but de mettre en place un certain nombre de tableaux de bord concernant les projets informatiques de la CDC. Ces tableaux font apparaître aux décideurs des informations pertinentes sur les projets informatiques qui porteront sur les coûts, les délais et les risques. L'objectif consistait à définir et mettre en place un ensemble de modules se greffant sur l'intranet de la Caisse des Dépôts. Ces modules constituent des interfaces d'alimentation et de consultation de la base de données de l'application RPI.

Estimations théoriques RPI

Grâce aux méthodes analysées, cette application RPI (ou tout du moins son premier lot) a pu faire apparaître les données suivantes :

Points de Fonctions Bruts : 73 ; Points de Fonctions Ajustés : 75.

Dans le cas d'un projet « simple », la méthode conseille d'utiliser une estimation 2 Jours*Homme par Points de fonctions. Soit une charge de $2 * 75 = 150 \text{ J*H}$

Méthode	Valeur nominale	Valeur pondérée
Réel	48	55
Probst	60	60
ESTIMAT	70	72
RAD - Evalueur	66	96
Points de fonction	146	150
COCOMO	281	331,5

Taille réelle de l'application : 3.798 KLOC, dont 2.500 KDSI (Chiffre donné par la MOE)

Taille théorique grâce aux Points de fonction :

$$T = 0.1187 * 75 - 6.49 = 2412 \text{ KDSI}$$

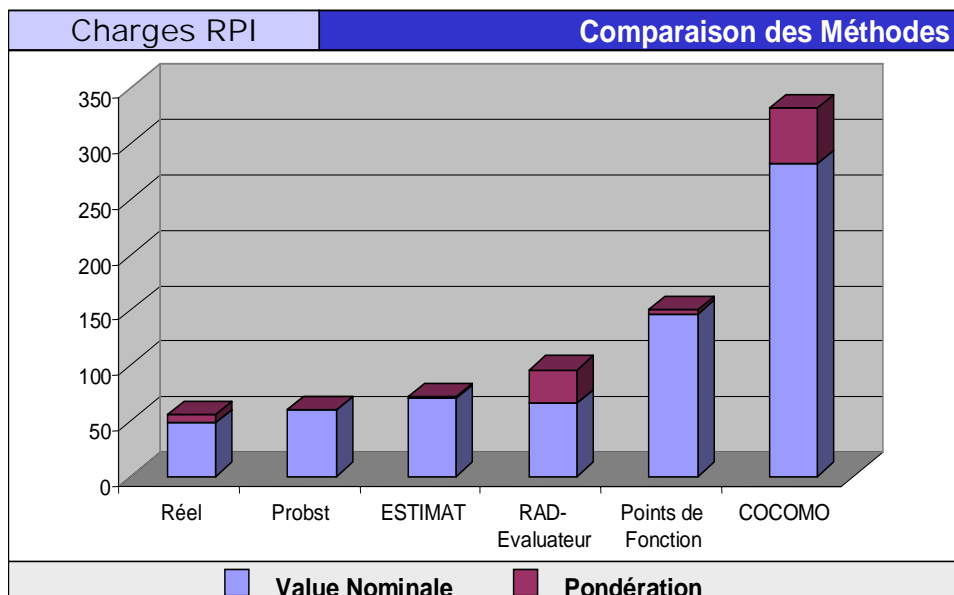
(Proche des 2500 réels)

COCOMO semi-détaché donne $E = 331.5 \text{ J*H}$

Évaluateur-RAD donne une valeur nominale du projet est de 66 J*H , ajustés à 96 J*H .

La méthode Probst fournit un résultat de 60 J*H

La méthode ESTIMAT fournit un résultat de 72 J*H



Le lot 1.1 de l'application RPI avait été estimé à 48 J*H . Sa réalisation en a consommé 55 .

On peut tirer de ces résultats les conclusions suivantes :

La méthode COCOMO ne semble pas être adaptée : si l'on considère réellement les KDSI ou les KLOC, en fonction du langage utilisé, pour la même application, on obtiendra des résultats réellement disproportionnés. D'autant que la méthode COCOMO ne tient pas du tout compte de l'application du côté client et de la complexité de l'ergonomie.

La méthode des points de fonction semble être excellente pour mesurer la « taille » d'une application, en terme de KDSI ou KLOC, mais le facteur de 2 JH par Point de Fonction est beaucoup trop élevé. Il aurait fallu, pour avoir un résultat exact, faire le calcul avec 0.75 JH par Point de Fonction.

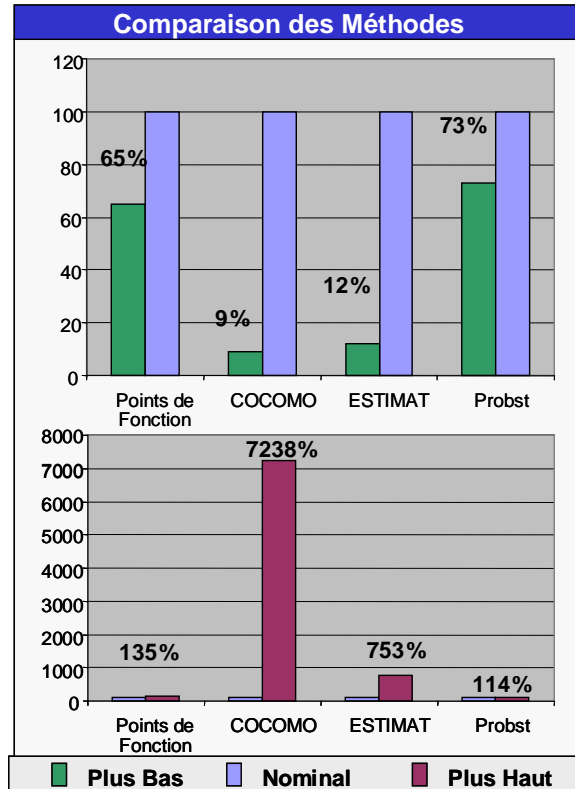
La méthode RAD-ÉVALUATEUR donne un résultat nominal (métrique d'origine) très acceptable, mais les pondérations opaques faites par le logiciel font perdre de la crédibilité à la méthode, ce qui est d'autant plus dommage que le logiciel a été développé spécifiquement pour estimer la charge des xNet.

La méthode ESTIMAT est proche du résultat final. Il faudrait probablement pour être proche de la réalité, y ajouter certaines données relatives aux modules d'identification et de navigation, et modifier les coefficients de complexité des composants.

C'est la méthode Probst qui se rapproche le plus de la réalité (moins de 10% d'erreur sur ce projet). Comme suggéré lors de l'analyse préalable, il faudrait là aussi, ajouter des données relatives aux modules d'identification et de navigation, et les charges d'opérations transverses, pour obtenir un résultat réellement fiable.

Conclusion

Au bilan, cinq méthodes plus ou moins différentes les unes des autres. Chacune a ses propres paramètres, chacune donne ses propres résultats.



Et la fourchette des paramètres entrés étant réellement large, on ne peut pas réellement faire ressortir les paramètres de métriques principaux. De plus, les variables de sortie étant elles mêmes peu cloisonnées, on ne sait jamais exactement ce que l'on est en train d'estimer (les tests unitaires et la recette sont-ils compris dans les charges de la réalisation, etc.)

En bref, vouloir faire ressortir un résultat précis et bien délimité, à partir d'une science non exacte prenant en compte toute sorte de paramètres, dont certains sont plus ou moins farfelus, se révèle être, au final plus un exercice de style, qu'une méthode fiable : on jongle avec beaucoup de choses, et il est on ne peut plus aisé de faire rendre à la méthode les chiffres que l'on veut.

Cependant tout n'est pas perdu : le fait de « croiser » toutes ces méthodes peut permettre de donner une estimation plus ou moins réaliste (à un facteur près, puisque toutes les évaluations se révèlent être trop élevées.)

Ainsi, une bonne solution pourrait être, si l'on désire effectuer rapidement une évaluation d'un projet xNet, de créer une sorte de « Tableau de Bord », destiné à la maîtrise d'ouvrage, qui serait également fourni à la maîtrise d'œuvre, et contenant un ensemble de points sur lesquels ils pourraient se mettre d'accord, en l'occurrence sur les paramètres d'entrée... ceci nécessitant une demi-journée, tout au plus.

La maîtrise d'ouvrage pourrait rapidement avoir un aperçu global des estimations ainsi faites.

Ce tableau de bord pourrait contenir les informations suivantes :

Estimation de la maîtrise d'œuvre, métriques des cinq méthodes (Nombre et complexité d'écrans, de tables, de fonctions, etc.), estimations faites avec les Points de Fonctions, COCOMO, RAD-Evaluateur, Probst, et ESTIMAT.

Présentation des marges d'inexactitude des pondérations en fonction des méthodes de la figure ci-contre. En effet, ces pondérations font varier la taille nominale du projet de façon inquiétante. La maîtrise d'ouvrage, en voyant ces chiffres, aura probablement le réflexe d'utiliser la méthode où l'erreur due à la pondération est la plus faible.

... cette solution amènerait la maîtrise d'ouvrage à juger elle-même du chiffre d'estimation des charges qu'elle peut obtenir, tout en lui mettant en avant qu'il est évident que le résultat obtenu sera d'une fiabilité relative, en raison des marges d'erreurs, et de la difficulté à comparer ces méthodes hétéroclites... Le plus important de ce « tableau de bord » étant probablement d'apporter des arguments nécessaires à une discussion saine et constructive, en matière d'élaboration de planning, entre les maîtrises d'œuvre et d'ouvrage.

Georges Zadrozynski

gz@msit.org

<http://www.gezzed.net>

Ingénieur EISTI

Mastère Management des S.I., HEC & ENSMP

Bibliographie

- La mesure du logiciel (Henri Habrias) 2ème édition revue, corrigée et augmentée, Teknea (ISBN : 2877170454)
- Piloter les projets informatiques de la nouvelle économie (Jean-Pierre Vickoff), Éditions d'Organisation (ISBN : 2708124870)
- Project Management: a Managerial Approach (Jack R. Meredith, Samuel J. Mantel Jr.), John Wiley and Sons (ISBN: 0471434620)
- Web Site Project Management (Ashley Friedlein), Morgan Kaufmann (ISBN: 1558606785)
- Gestion d'un projet système d'information - Principes, techniques et mise en œuvre (Chantal Morley), Dunod (ISBN: 2100059734)
- Précis de Conduite de Projet Informatique (Cyrille Chartier-Kastler), Éditions d'Organisation (ISBN : 2708118145)
- Conduite de projets informatiques : principes et techniques (José Moréjon, Jean-René James), Interéditions (ISBN : 272960457X)
- Le management d'un projet (H.-P. Maders), Éditions d'Organisation (ISBN : 2708117947)
- Que Sais-je : Le management de projet (Olivier Badot, Jean-Marie Hazebroucq), Presses Universitaires de France (ISBN : 2130474179)
- Le maître d'ouvrage du système d'information (Alain Berdugo), Hermès Sciences (ISBN : 2866016211)
- Guide du Management des Systèmes d'Informations - Thèmes & Termes Essentiels (Alain Berdugo, Robert Mahl, Gérard Jean, HEC/ENSMP MSIT 2002), Hermès Sciences (ISBN : 2746205246)
- De la Stratégie aux Systèmes d'Information (ALTIME) Formation ALTIME/HEC/ENSMP, février 2002
- [ADELI] Estimations de charges - Les orientations de la commission (Alain Coulon)
Lettre ADELI n°39 - Square des Utilisateurs - avril 2000
- [ADELI] Estimation de projets informatiques (Kathleen Peters, traduit par Alain Coulon)
Lettre ADELI n°41 - Square des Utilisateurs - octobre 2000
- [ADELI] Calibrage et étalonnage (Jean Joskowicz)
Lettre ADELI n°42 - Square des Utilisateurs - janvier 2001

[ADELI] Science ou magie ? - Le point sur les estimations des projets logiciels (Nicolas Trèves et Alain Coulon)
Lettre ADELI n°45 - Square des Utilisateurs - octobre 2001

La méthode des Points de Fonction (Olivier Denel), Thèse MSIT (HEC/ENSMP) -
<http://www.denel.com/pdf/index.php>

Rapport CIGREF : Réseaux Internet/Intranet, Rapport CIGREF, septembre 1997

COCOMO: Resource Estimation (David Stotts, Associate Professor, Dept. of Computer Science)
University of North Carolina - <http://www.cs.unc.edu/~stotts/COMP145/cocomo.html>

IFT3902 : Développement, maintenance de logiciels (Université de Montréal : Faculté des Arts et des Sciences)
Informatique et recherche opérationnelle,
<http://www.iro.umontreal.ca/~pift3902/Cours2001/Planification.pdf>

Estimation des coûts et délais par la méthode COCOMO (Tina Wilhelm, Évelyne Parthenay, Hugues Am)
École Supérieure en Sciences Informatiques,
<http://www.essi.fr/~hugues/GL/COCOMO/cocomo.html>

Algorithmics Costs Model (Dan Snell), Bournemouth University - <http://www.ecfc.u-net.com/cost/models.htm>

Évaluation, la théorie (Jean-Pierre Vickoff), <http://mapage.noos.fr/rad/boudeval.htm>

Principes d'Évaluation (Jean-Pierre Vickoff), <http://mapage.noos.fr/rad/evalint.htm>

Méthode Interne CDC – Feuille de Calcul d'Estimation de Projets (André Probst)

Méthode ESTIMAT (Informatique CDC, Méthodes Ingénierie Progiciels)



Histoires de rasoirs (1)

Le rasoir d'Ockham

Première histoire de rasoir, dont le rapport avec les systèmes d'information est moins éloigné qu'il n'y paraît à première vue.

Frère Guillaume

Né en Angleterre vers 1285-1290, Guillaume d'Ockham entra très jeune dans l'ordre des Franciscains. Il enseigna la théologie à Oxford, passa plusieurs années à Avignon, à défendre ses idées auprès du pape Jean XXII, puis mourut, probablement de la peste noire, en 1349 à Munich où il s'était réfugié auprès de Louis de Bavière.

Ce franciscain libre penseur contestait l'infaillibilité du Pape et des conciles. Il vit plusieurs de ses propositions condamnées comme hérétiques et prit parti contre le Pape pour l'empereur Louis de Bavière dans la question du pouvoir temporel de l'Église. Ceux qui ont lu « Le Nom de la Rose » d'Umberto Eco reconnaîtront Guillaume de Baskerville, qui fut interprété au cinéma par Sean Connery¹.

Guillaume d'Ockham est aujourd'hui reconnu comme un grand philosophe et logicien et, ce qui allait de pair à l'époque, comme un grand méthodologue. Ce n'est pas un hasard si nombre de scientifiques se réfèrent aujourd'hui à Ockham pour aborder les sciences de la complexité.

La doctrine de Guillaume d'Ockham est qualifiée de nominaliste : elle soutient que seules les choses particulières appartiennent à la réalité, tandis que les choses universelles ne sont que des idées abstraites. Des notions telles qu' « arbre » ou « homme » n'existent que dans le monde des idées, le monde réel est peuplé d'individus distincts, qui peuvent être qualifiés du nom d' « arbre » ou d' « homme » en fonction de certaines qualités apparentes.

Guillaume d'Ockham est surtout connu pour le fameux principe d'économie, passé à la postérité sous le nom de « rasoir d'Ockham », « pluralitas non est ponenda sine neccessitate ». « Il ne faut jamais poser une pluralité sans y être contraint par la nécessité » ou comme on l'énonce souvent : « il ne faut pas multiplier les êtres sans nécessité ».

Ainsi, pour expliquer un phénomène qui se manifeste à notre expérience, on doit s'efforcer d'en trouver la ou les causes dans d'autres phénomènes que nous connaissons déjà et on n'aura recours à des principes supérieurs, tels que la volonté divine ou le miracle que si toute autre explication a échoué. Ockham juge économique d'admettre que, dans la nature, Dieu laisse agir les causes physiques naturelles sans intervenir lui-même sans cesse. Par exemple, c'est bien le feu et non Dieu ou le Diable qui brûle. On peut donc utiliser l'induction et la déduction pour connaître. Ceci marque le début de la méthode expérimentale.

Ockham pourchasse ainsi dans les moindres recoins de la philosophie et de la théologie les pseudo-essences et pseudo-causes que ses prédécesseurs avaient inutilement multipliées.

C'est ce principe qui, au XVI^{ème} siècle, poussa Copernic à choisir la théorie héliocentrique du système solaire, et à refuser de croire en la théorie dominante de l'époque, le géocentrisme qui plaçait la terre au centre du système, alors que les deux théories se voulaient une explication des mêmes données observées. Les orbites régulières et symétriques que le modèle héliocentrique de Copernic attribuait

1 Dans le roman, comme dans le film, un livre sur la comédie est la cause des meurtres qui déciment une abbaye bénédictine dont les moines sont traducteurs et copistes de textes antiques. Ce livre est le second tome - perdu - de La Poétique d'Aristote qu'un moine aveugle et fou veut empêcher quiconque de lire, sous le prétexte que le rire tue la peur et provoque ainsi le plus grand désordre.

aux planètes étaient bien plus séduisantes intellectuellement. Sur la base de ce seul principe, Copernic se fit l'apôtre de l'héliocentrisme et prêcha contre le géocentrisme. Entre deux théories qui permettaient de prédire exactement les mêmes mouvements, il choisit la plus simple.

Le rasoir d'Ockham se retrouve sous différentes formes :

- principe d'économie ;
- principe de la moindre action ;
- principe de parcimonie ;
- principe de simplicité ;
- principe de paresse ;
- principe du moindre effort ;
- plus court chemin entre deux points.

J'ai voulu en avoir le cœur net : Ockham n'a jamais évoqué le moindre rasoir dans ses œuvres. Il semblerait que l'expression « rasoir d'Ockham » ait été inventée ultérieurement par un autre philosophe du Moyen-Âge, Buridan².

Quel rapport entre Ockham et les systèmes d'information ?

Ockham a donné son nom à un langage de programmation, Occam, développé en Grande-Bretagne par David May. Le premier objectif de ce langage, permettant une programmation en parallèle, était comme on peut le supposer la simplicité.

De façon plus générale, le principe de simplicité devrait naturellement guider la démarche de conception de nos systèmes. L'application du rasoir d'Ockham devrait nous permettre d'élaguer les fonctionnalités inutiles de nos logiciels obèses³ et de supprimer les entités superflues de nos modèles de données.

On peut considérer, à juste titre, que l'analyse de la valeur est une application directe du rasoir d'Ockham.

La simplicité s'oppose à la complication, elle permet d'appréhender les systèmes complexes.

Ne multiplions pas les entités sans raison.

Cette maxime s'applique quotidiennement dans la vie de l'organisateur, du concepteur ou du développeur. On pourrait la décliner sous des formes multiples, telles que :

Ne multiplions pas les procédures

Ne multiplions pas les niveaux hiérarchiques

Ne multiplions pas les intermédiaires

Ne multiplions pas les signatures

Ne multiplions pas les destinataires de nos courriels

À vous d'en trouver d'autres !.

Martine Otter
Présidente d'ADELI

² Recteur de l'Université de Paris en 1328 puis en 1340, Jean Buridan, a laissé de nombreuses fables attachées aux traditions populaires, dont l'âne de Buridan représentant quelqu'un qui ne sait pas se décider, car l'âne pressé par la faim et la soif se laissa mourir entre une mesure d'avoine et un seau d'eau...

³ Voir l'article d'Yves Constantinidis - Qualité et quantité / Logiciel élégant contre logiciel obèse - Lettre n°45 - Octobre 2001



Histoires de rasoirs (2)

Quoi de neuf chez Gillette ?

Deuxième histoire de rasoir, dont le rapport avec les systèmes d'information est assez évident, et le rapport avec la première histoire de rasoir plus évident qu'il n'y paraît à première vue.

Innovation dans le monde du rasoir

Comme beaucoup d'entre vous je consacre tous les jours quelques minutes à parcourir d'un œil distrait les dernières nouvelles envoyées sous forme de mailing-list par quelques sites d'information, tels que 01net, ZDNET, ou autre JDNET news.

Lors de cette veille technologique quotidienne, mon regard fut attiré le 31 janvier dernier par une curieuse information, suffisamment insolite pour me donner l'envie de m'y intéresser : sous le titre « Gillette, le rasoir qui espionne de près »⁴ 01 nous signalait que « Des puces capables d'émettre des informations vont être incorporées dans 500 millions de rasoirs. Les associations militant pour le respect de la vie privée dénoncent déjà les dérives que cela peut entraîner. »

Je ne vis nulle part cette information reprise par la presse et me demandais s'il ne s'agissait pas d'un canular, poisson d'avril un peu prématuré.

Cet article faisait suite à un article paru quelques jours plus tôt, le 17 janvier, « Logistique : Gillette teste le marquage intelligent ». L'objectif clairement énoncé était, par l'introduction d'un microprocesseur à l'intérieur de l'emballage, d'assurer le suivi des marchandises, depuis leur fabrication jusqu'aux linéaires des commerçants. La technologie utilisée, émission de signaux radio à partir de minuscules puces, RFID (Radio Frequency Identifiants), mise au point par le MIT, existerait depuis quelques années et intéresserait quelques industriels fabricants de produits grand public, tels que Philip Morris ou Procter & Gamble.

Gillette, aurait acheté 500 millions de tags RFID pour en équiper ses produits, ceci lui permettant officiellement d'assurer le suivi des stocks et d'évaluer la rotation de ses produits dans les magasins via cette technologie. L'usage anti-vol n'est pas directement évoqué mais nous savons que les rasoirs font partie de la liste des objets les plus volés dans les grandes surfaces. D'après certaines estimations, plus de 5% des Gillette Mach3 en circulation ont été volés, et peut-être près de 50% des Mach3 vendus hors des réseaux classiques de distribution (sur eBay ou auprès de vendeurs peu scrupuleux) pourraient « être tombés du camion »...

Dans les faits, le Mach3 est sans doute l'objet le plus volé en Europe, pour ne pas dire dans le monde, et c'est donc le candidat parfait pour tester les systèmes antivols.

La technologie RFID

L'identification par radiofréquence, autrement dit la RFID, est une technologie qui basée sur la transmission par ondes radio, permet d'identifier de manière fiable et unique un objet, un animal ou une personne par une étiquette électronique capable d'émettre des données. Elle apparut dans les années 80 dans des applications de traçabilité, en remplacement des étiquettes à code barres, dans des environnements où celles-ci étaient difficilement utilisables. Les systèmes RFID se sont développés en particulier pour la gestion des stocks et l'identification automatique des véhicules en mouvement.

⁴ <http://www.01net.com/article/200931.html>

Un système d'identification par radiofréquence est composé d'une part de l'étiquette radio contenant l'information (un numéro unique et une mémoire) et un lecteur / encodeur se connectant au système informatique.

L'étiquette est activée par le signal radio de fréquence déterminée puis émet un signal à son tour. Le lecteur, lui-même composé d'une carte électronique et d'une antenne, se comporte alors comme un émetteur / récepteur : il lit et peut écrire des informations dans la mémoire de la puce.

Le RFID utilise des technologies Internet. Equipé adéquatement, votre rasoir, ou votre canette de Coca, transmet un code d'identification unique lorsqu'il est interrogé par un lecteur. Cet identifiant fonctionne comme une URL et est traduite en adresse IP par une base de données DNS, un peu comme une adresse Web. L'adresse IP est reliée à des informations sur l'objet dans la base de données.

Ainsi, où que soit l'objet dans le monde, vous pouvez tout savoir à son sujet et il suffit pour cela d'avoir un identifiant unique sur l'objet. S'il est volé, cela déclenche une alarme lorsqu'il est à quelques mètres d'un lecteur.

Les systèmes, fabriqués par Siemens, Texas Instruments, Philips, ou encore Motorola se sont progressivement miniaturisés, passant de la taille d'un ongle dans les années 80 à celle d'un grain de sable aujourd'hui. Ils ont accru leur portée de saisie à distance qui peut atteindre désormais une dizaine de mètres. Leur capacité de stockage maximum est actuellement de 64 Kb, soit beaucoup plus que ne pouvait en contenir une simple étiquette code barres.

A quoi ça sert ?

Le RFID serait issu de l'invention du radar faite lors de la seconde guerre mondiale. Son utilisation commerciale a commencé dans les années 70 pour identifier le bétail. En France, une des dernières utilisations concerne le passage des péages autoroutiers sans s'arrêter. Dans le monde de l'industrie, l'utilisation du RFID s'est largement répandue dans les années 80. Les constructeurs automobiles ont commencé à l'utiliser sur leurs chaînes de production.

La RFID fait partie de la famille des technologies de la traçabilité. Ses clients sont la grande distribution, l'automobile, les laboratoires. Les fournisseurs s'adressent également au monde des armées pour la Logistique du « Matériel » militaire sur zone, le repérage topographique militaire par cartographie mobile en liaison GPS. Ou les Inventaires mobiles temps réel. Les administrations s'intéressent de leur côté à la gestion des bibliothèques et l'entreposage des documents papier.

Les applications sont nombreuses :

- système antivol, ;
- protection contre la contrefaçon ;
- gestion des supports de manutention ;
- contrôle des expéditions ;
- contrôle des réceptions ; ...

Selon le *RFID Journal*, Michelin aurait prévu d'implanter des puces dans chacune de ses roues pour transmettre des informations sur leur état (pression, usure...) par fréquences radio à l'ordinateur de bord. De son côté, la société allemande KSW-Microtec aurait inventé un tag RFID lavable, prévu pour être incorporé à des vêtements.

Le cabinet Frost & Sullivan estime qu'en 2004 ce marché devrait représenter au niveau mondial un chiffre d'affaires de 277,82 millions de dollars. Toujours selon Frost & Sullivan, le marché mondial pour ses applications naissantes se développe à un taux de croissance annuel d'environ 33 %. Les revenus qui seront générés d'ici les deux ou trois prochaines années seront issus d'applications très variées dont la liste s'allonge tous les jours :

- grande distribution pour le marquage des produits frais (Isigny, Charal pour la viande) ;
- blanchisserie industrielle pour le marquage résistant des vêtements et textiles (Elis) ;
- distribution de gaz pour la gestion des bombonnes et des opérations de remplissage (Butagaz, Air Liquide) ;

- ou encore industrie automobile pour le marquage des véhicules (Renault, Fiat, Gefco,...) ou des pneus montés afin de les rappeler en cas de besoin⁵.

Aux dernières nouvelles⁶, également relayées par 01, Benetton envisagerait d'incorporer des puces RFID dans les vêtements de sa ligne Sisley et passerait commande à Philips de 15 millions de puces livrables en 2003.

Menace pour la vie privée ?

Depuis la crise de la « vache folle », le mot traçabilité évoque pour le grand public, la sécurité alimentaire. Il est essentiel de connaître la provenance des aliments que nous consommons. Il est également essentiel de prendre conscience du fait que les technologies développées pour la traçabilité alimentaire peuvent être aussi bien utilisées pour le marquage de tout objet ou de tout être vivant.

Traçabilité cela suppose de « suivre à la trace ». La traçabilité des objets peut dériver facilement vers la traçabilité des personnes porteuses de ces objets, ceci à leur insu. James Bond n'est pas loin.

Des techniques de désactivation simple de la puce seraient à l'étude et pourraient permettre de neutraliser la puce à la demande du client lors de son passage en caisse. Mais il est trop tôt pour savoir si ces systèmes de désactivation pourront être mis en œuvre. Les fabricants n'ont pas forcément intérêt à désactiver ces puces à la sortie du magasin, car elles constituent en fait une sorte de tatouage les protégeant de la contrefaçon.

Les premières réactions un peu épidermiques sur le forum de 01 proposaient de boycotter les rasoirs Gillette. Et peut-être quelques centaines d'informaticiens vont-ils réellement boycotter Gillette, voire se laisser pousser la barbe. Cela n'aura pas grand effet sur le développement de la technologie RFID dont le grand public ignore presque jusqu'à l'existence et dont les associations françaises de consommateurs semblent ne pas avoir entendu parler. Je me demande d'ailleurs, sans trouver d'hypothèse satisfaisante, pourquoi la presse informatique s'est spécialement intéressée à la technologie RFID à l'occasion de cette histoire de rasoir.

Quelles perspectives ?

Le principe de traçabilité est un principe de précaution : chaque objet est unique, on ne peut pas parler de la viande de bœuf en général, ni même de la viande de bœuf, anglaise. Chaque animal est unique. Toute généralisation doit être écartée. Il convient donc de marquer tous les objets si l'on veut connaître précisément leur provenance ou leur localisation. Et c'est bien ce que fait Gillette, en marquant individuellement chacun de ses rasoirs. Un lot trop coupant pourrait ainsi être repéré et retiré de la vente : je doute cependant que cela soit l'objectif réel du marquage des rasoirs.

Il s'agirait plutôt de la part de Gillette d'une opération de marketing « one to one » : chaque client est unique, les catégories statistiques n'existent pas plus que le français moyen ou la ménagère de plus de 50 ans, dont on ne sait pas si elle achète les rasoirs de son mari ou les utilise comme outil d'épilation. Gillette a-t-il l'intention d'affiner sa connaissance de l'acheteur de rasoir ? Ou seulement de pister les voleurs ? Sans doute, un peu des deux, l'objectif principal dans les deux cas étant la maximisation du profit.

Difficile de conclure. Les outils ne peuvent pas être jugés dans l'absolu mais bien au regard de l'usage qui en est fait. Le marteau comme la main peuvent également détruire ou construire, avec toutefois une force différente. La technologie RFID offre des avantages certains en matière de traçabilité et donc de prévention en matière d'hygiène alimentaire et de santé. En s'appliquant de façon non contrôlée à des données personnelles, elle peut également menacer le respect de la vie privée. Il revient à chacun d'entre nous de contribuer aux décisions qui seront prises, à la fois en tant que citoyen utilisateur des systèmes d'information et en tant que professionnel à l'écoute des besoins des utilisateurs.

Martine Otter
Présidente d'ADELI



Changement de GENRE ! ...

Pastiche d'un Extrait de La Lettre d'ADELI N°29 - Octobre 1997

Nous avons testé Époux V1.0. Dans le cadre des travaux pratiques de la commission GCL¹, il était important de vérifier l'application de concepts de symétrie à la gestion documentaire. Ce petit travail en est l'illustration.

On ne devrait jamais modifier un environnement de travail qui vous donne satisfaction !

Tenez, l'année dernière, par exemple, une de mes amies, bien que pleinement satisfaite des services que lui rendait PETIT-COPAIN (version 1.0) a pourtant décidé (attrait de la nouveauté ? crise de déprime ? allez savoir) de sauter le pas et de s'offrir directement l'upgrade complet, à savoir ÉPOUX (toujours version 1.0).

Elle a rapidement découvert qu'ÉPOUX 1.0 accaparait presque toute sa mémoire et ne laissait que très peu de place pour les autres applications.

De plus, elle vient de s'apercevoir qu'ÉPOUX 1.0 était maintenant entré dans un processus de reproduction², sur une durée imprévisible³, et que ce phénomène entraînerait, quoi qu'il advienne, un nouveau et important gaspillage de ressources système et un lourd fardeau à porter.

Bien entendu, aucune mention de cet étrange comportement ne figurait dans la documentation dont il disposait au moment de l'achat.

D'autres utilisatrices l'avaient cependant averti que ce risque existait, en raison, disaient-elle, de la nature même de cette application.

Le pire en définitive, c'est qu'ÉPOUX 1.0 s'installe de manière telle à toujours démarrer en premier, dès l'allumage du système, ce qui lui permet non seulement de contrôler toutes les activités qui se produisent ensuite dans l'environnement, mais aussi d'interférer avec elles, voire même d'en empêcher purement et simplement le lancement !

Mon amie s'est en effet aperçue avec effroi que le système lui refusait dorénavant tout accès à diverses applications, comme COURS-de-DANSE-avec-les-COPINES⁴ (version 10.3), TOURNÉE-des-SALONS-de-THE (version 2.5), SHOPPING-tout-le-WEEK-END (version 7.0), MISE-en-NEGATIF-du-COMPTÉ-BANCAIRE-du-Ménage (version 15.2.7), applications auxquelles elle tenait énormément, et qui fonctionnaient jusque-là sans problème !

Si elle s'obstinait néanmoins à vouloir les lancer, le comportement du système tout entier devenait instable et même imprévisible, ce qui conduisait généralement à un crash complet avec la nécessité d'un allumage à froid, avec les conséquences que l'on devine aisément...

De plus, lors de l'installation, ÉPOUX 1.0 ne fournit malheureusement aucune option permettant de désinstaller les innombrables « plug-ins » et autres « add-ons » indésirables dont l'application fait par ailleurs grand usage (nouveau gaspillage de ressources système), tels BELLE-MAMAN 55.8 (qui occupe énormément de place) et BEAU-FRÈRE (celui-là hélas toujours en phase Bêta).

Pour couronner le tout, les performances globales du système semblent diminuer de jour en jour au point que mon amie envisage carrément de changer d'application !

Michel Demonfaucon

¹ La Gestion de Configuration Logiciel se préoccupe effectivement de la maîtrise des changements de version.

² Tel le mécanisme de fork d'Unix, le processus père engendrant un processus fils.

³ Il y a perte de symétrie par rapport au texte d'origine.

⁴ Bien qu'il y ait incompatibilité avec le processus de fork précédent.



Chemin de la Liberté

Vers une informatique durable et humaine ?

une vision (historique et vaste) de Bruno Lussato⁵ sur notre monde de l'informatique et sur son devenir

Vers une informatique « durable, humaine » ? Cela laisserait-il supposer que l'informatique planétaire actuelle ne le serait pas ? C'est ce que l'on pourrait comprendre de l'analyse de Bruno Lussato, l'un des conseillers les plus écoutés du monde des grandes entreprises sur les technologies de l'information et l'un des promoteurs de la micro-informatique. Une analyse très large dont chacun appréciera (ou non) la pertinence. Celle-ci rejoint les réflexions actuelles sur la nécessité impérieuse de passer à un « développement durable ».

Introduction

Nous avons eu l'occasion d'assister, en novembre 2002, dans un grand hôtel parisien, aux « Assises de l'économie connectée », à l'invitation d'un éditeur de progiciels⁶, organisateur de la réunion.

Devant un parterre d'entrepreneurs, Bruno Lussato y a présenté les idées rassemblées dans son ouvrage « La troisième révolution »⁷. Voici quelques-unes des idées majeures que nous en avons retenues au cours de cette conférence intitulée « une informatique à visage humain ». Que l'on nous pardonne de la schématisation qui suivra.

L'auteur observe que nous sommes à la veille d'une nouvelle révolution mondiale de l'informatique et des systèmes d'information. Cette révolution va bouleverser plus encore que les précédentes, les sociétés, les économies, les cultures et les hommes ; en quelque sorte, le destin de la planète.

Ce mouvement s'effectuera, selon l'état des influences et des rapports des forces en présence :

- soit dans une direction régressive et destructive ;
- soit dans une direction promotrice et évolutive ;

quant à ses conséquences sociales, économiques, humaines et environnementales.

Inévitable, cette révolution imminente (la troisième de l'aventure informatique^o) marquera une étape de l'histoire de la société.

Première révolution : avènement de l'informatique et concentration des organisations

Le passage de la mécanographie à l'informatique a déclenché la première révolution informatique et organisationnelle. Ce fut le saut, qualitatif et quantitatif, du passage de machines à composants mécaniques (tabulatrices, perforatrices, trieuses de cartes, etc.) à des machines à composants entièrement électroniques.

C'est la recherche militaire lors de la deuxième guerre mondiale, qui donna la première impulsion décisive de cette révolution. Puis, l'ordinateur Univac vint bousculer les fabricants de machines

⁵ Bruno Lussato, en 1974, fut le premier en France, et sans doute l'un des premiers dans le monde, à annoncer le triomphe de la micro-informatique sur l'informatique centralisée. Une bibliographie est présentée en fin d'article.

⁶ Assises de l'économie connectée, organisées par JD Edwards le jeudi 21 novembre 2002 à Paris, dans les Salons de l'Opéra.

⁷ « La troisième révolution » de Bruno Lussato, aux Editions Plon 1999 (l'ouvrage semblerait actuellement épuisé)

mécanographiques (dont IBM) sur leur propre marché avec une technologie à laquelle beaucoup d'acteurs de l'industrie ne croyaient pas encore.

Cette première révolution a engendré une concentration et une centralisation majeure, que l'on pourrait qualifier de napoléoniennes, dans les entreprises ; cela, essentiellement, pour des raisons de logique d'économie d'échelle (coût extrême des composants et des installations, rareté des experts matériels et logiciels, fortes contraintes de climatisations, etc.) On traitait alors en priorité les tâches les plus stables et les plus répétitives telles celles de la comptabilité. Les gains économiques réalisés ont été considérables, souvent au détriment des « employés aux écritures ». Ce fut le triomphe d'une logique de spécialisation des tâches transférée du domaine industriel au domaine de l'administratif: le service de l'ordinateur se transformant parfois un peu en « Les temps modernes » de Chaplin transféré au bureau !

Seconde révolution : l'avènement micro-informatique, des effets contradictoires

La seconde révolution informatique et organisationnelle identifiée par Bruno Lussato aurait été déclenchée par l'avènement de la micro-informatique (à laquelle il faudrait sans doute adjoindre, à notre sens, celui de l'informatique « départementale », qui donnera progressivement naissance aux « systèmes ouverts »).

La loi de recherche d'économie d'échelle (loi de Grosh) se trouva peu à peu invalidée par la loi de la chute du coût et du volume des composants (loi de Moore). Une partie de la puissance rendue ainsi disponible par l'abaissement des coûts fut canalisée, on s'en souvient, au service de l'accroissement de la convivialité de l'informatique et de l'interface homme/machine.

En parallèle, l'informaticien n'était plus une ressource rare, ce qui rendait caducs les modèles anciens de synergie.

Ces modèles ont continué néanmoins trop souvent, à servir encore, de façon anachronique, de guide d'action : ainsi a-t-on continué, par exemple, à économiser de la mémoire pour représenter des dates, là où cela n'était plus justifié et représentait même un danger pour l'an 2000, à développer des traitements de gestion en assembleur, complexifié des traitements informatiques par une logique batch et séquentielle quand ce n'était plus nécessaire...

Ce fut le triomphe de la logique de la mini et de la micro-informatique. Mais, les leaders mondiaux du marché des ordinateurs, de culture centralisatrice, n'en ont pris que tardivement la mesure, ce qui fit le profit de Microsoft et d'Intel.

On peut identifier deux types de conséquence à cette révolution.

Les aspects positifs : rapprochement de la machine de l'homme

Le surcroît de puissance et d'intelligence humaine, disponible à moindre coût, permet de traiter l'information non plus seulement quantitative mais également dans sa partie qualitative (interfaces graphiques, logiciels spécifiques, informatique créative voire ludique). De plus, elle permet de mettre en œuvre des systèmes « d'intelligence artificielle » plus complexes tels que les calculs météorologiques. Les nouveaux systèmes déconcentrés, ainsi rendus possibles, se révèlent plus conviviaux et moins vulnérables.

Les aspects négatifs : adversaires et freins à la déconcentration

L'entreprise ne semble guère, prioritairement et fondamentalement, demandeuse d'informations qualitatives, en dépit des discours d'intention. Elle exige, au contraire, en priorité du quantitatif grâce à une réduction, en coûts et en délais des traitements de l'information.

La chute, à performance constante, du volume et du prix de l'informatique, concernant le matériel mais aussi le logiciel, menace constructeurs et informaticiens internes. Selon Bruno Lussato, la coalition du lobbying « centralisateur », financier, comptable, informaticien ou fournisseur, se dresse alors souvent contre une déconcentration, laquelle heurte culture et impératifs à court terme, et

encourage la construction de Systèmes d'Information « gratte-ciel » plus grands consommateurs en données à traiter⁸.

On aboutit donc, dans cette logique centralisatrice dominante, à des systèmes d'information de plus en plus opaques, spécialisés, vulnérables, standardisés, éloignés de l'humain. Cela en remplaçant la loi de Pareto (consistant à l'économie de flux et de stocks d'information) par la logique de puissance et de banalisation.

Le traitement qualitatif de plus en plus élaboré tend en même temps à être absorbé dans l'entonnoir du traitement quantitatif.

Les fréquentes fusions d'entreprises renforcent encore cette tendance à la concentration et à l'opacité des systèmes d'information.

Troisième révolution : bipolarisation entre MATRIX et REFONDATION

La troisième révolution informatique et organisationnelle serait aujourd'hui, pour Bruno Lussato, encore à l'état de prémices, en donnant des signes extrêmement clairs, mais sans que celle-ci soit encore réellement engagée.

Elle serait constituée par une lutte impitoyable entre :

- d'un côté, un principe dominant de massification, de centralisation et de mondialisation financière, qui aurait entamé son déclin ;
- et, de l'autre côté, un principe ascendant de dé-massification, de décentralisation et de démultiplication des initiatives entrepreneuriales individuelles et collectives.

Par analogie et en hommage à des œuvres culturelles « poétiques » et philosophiques connues, l'auteur nomme « MATRIX » le premier principe, « REFONDATION » le second.

MATRIX : organisation terrifiante moderne ou dépassée ?

MATRIX remplacerait la réalité par le virtuel, les hommes par des « strates », des numéros, des cibles, l'intuition par des logiciens, la philosophie des entreprises par la langue de bois.

MATRIX serait fondé sur le postulat archaïque de synergie et économie d'échelle, de banalisation et de mondialisation, de concentration et d'abstraction.

Ce système diminuerait globalement les richesses créées, la valeur ajoutée et les marges de chacun, ne pourrait subsister que par une « prédation » sauvage et anarchique. Il ne serait socialement plus compétitif mais au contraire ruineux pour la société, pour l'homme et pour la planète et son environnement⁹.

Les conditions qui ont justifié ce type d'organisation, à une certaine époque, ont d'ailleurs disparu.

Mais, sous la bienveillance de grands cabinets d'audit et de conseils que l'on ne citera pas (cf. l'actualité récente), nous vivons l'époque des multiples « retours de bâton » que l'on connaît : scandales en bourse, chute de la confiance dans la grande entreprise, public grugé et petits porteurs victimes de cette « prédation ».

Au plan culturel, mental, à l'aide de médias dominateurs, le gouffre semblerait aussi vouloir être agrandi entre, d'un côté, la masse la plus importante de cerveaux que l'on voudrait « abêtis », banalisés et, de l'autre, une élite de plus en plus réduite, « initiée », informée, privilégiée : ainsi des employés hautement qualifiés ou des responsables d'encadrement transformés peu à peu en exécutants « sans âme ».

⁸ Cette analyse, présentée ici de façon simplifiée et schématique, techniciste, serait certainement un peu à affiner. Effectivement, une certaine tendance au « downsizing », promue alors par des fournisseurs et acteurs économiques majeurs du marché, a été parfois plus « gourmande » en traitements d'informations inutiles qu'une vision plus sagement fédératrice, comme on s'en souvient.

⁹ NDLR : Cette vision n'est pas sans rappeler celle du « Sommet de la Terre » tenu au Brésil et des sommets similaires précédents.

Au plan matériel aussi, l'Internet et les jeux vidéo, par exemple, loin de diminuer cette bipolarisation (comme l'avaient annoncé à tort maints penseurs en vue presque « officialisés » voilà vingt ans) approfondi le fossé culturel sur le plan mondial comme sur le plan local.

Mais, pour Bruno Lussato, la technologie, évoluant très rapidement, comme la langue d'Ésope, peut le pire, en servant MATRIX, comme le meilleur, en étant l'alliée de REFONDATION, une organisation à visage humain fondée sur la différence et la rappropriation de l'information et son traitement par chacun.

REFONDATION : une organisation durable à visage humain

De même que naguère, la micro-informatique et l'informatique répartie ont progressivement remplacé les mainframes (même si ceux-ci gardent encore une place importante), REFONDATION devrait demain prendre naturellement la domination sur MATRIX, ceci pour trois raisons : une économique, une commerciale, et une humaine.

- Son premier atout est économique : les méthodes héritées du ré-engineering et autres « budget base zéro » permettront de diviser les coûts informatiques par un facteur supérieur à dix, à performances égales.
- Son second atout est commercial : en aucun cas des logiciels basés sur des modèles préétablis « ne peuvent absorber l'incertitude et la variété du monde, ni affronter la diversité croissante des besoins », ce que, seul, l'homme est à même de proposer et de « vendre » sur le marché. Il est donc économiquement indispensable de rapprocher la décision du terrain, y compris la décision informatique, face à une logique de superficialité de décision des empires (cf. récentes chutes de géants).
- Son troisième atout est humain : une entreprise « d'hommes responsables, épanouis, maîtres de leur information, autonomes par rapport aux décisions d'un siège lointain, l'emportera toujours et triomphera sur une entreprise de fantômes résignés, apeurés ou, pire encore, satisfaits de leur sort, dominés par le système en place et quelques cerveaux paranoïaques.

Quelle finalité donner à l'informatique ?

Dans cet « affrontement », il est un fait que l'informatique joue un rôle central. C'est elle, en quelque sorte, qui aurait contribué à façonner une organisation de type MATRIX, et c'est donc à elle, à présent, qu'il incomberait de servir une organisation de type REFONDATION. C'est cela que l'auteur nomme donc dans son livre la « troisième révolution ».

Pour cela, il reviendra à des entreprises novatrices et à des femmes et des hommes prêts, désireux et motivés, de montrer l'exemple aux autres et d'ouvrir les fenêtres nécessaires, anticipant l'avenir et le préparant, revalorisant pour cela « l'esprit entrepreneurial, le bas coût administratif, le tissu humain ».

Les entrepreneurs, les informaticiens ont toute leur place dans cette tâche. Mais celle-ci sera très difficile, car il faudra mettre en place des « méthodes de simplification des processus » qui ne seront pas influencés par la logique MATRIX culturellement dominante. Et il conviendra, en outre, de trouver les voies de la transition...

Comment changer de culture et d'organisation ?

Passer de l'ancienne culture et de l'ancienne organisation aux nouvelles, sera une transition difficile. Comment cela peut-il s'effectuer ?

L'organisation REFONDATION utilise « des outillages simples issus de l'informatique industrielle, des communications aussi réduites que possible, un centre de gravité très bas ». Elle remplace l'informatique « distribuée » par l'informatique « répartie » (individualisée, personnalisée).

Cette réorganisation exigera donc des compétences très solides et très motivées : re-engineering informatique, modélisation des processus d'activité et des flux d'information, expertise réseaux et systèmes ouverts, organisateurs... Et puis « des gens de terrain, encore des gens de terrain, toujours des gens de terrain ».

Ces compétences et cette volonté ne seront donc pas automatiquement à la portée de chacune et chacun, ni de chaque entreprise (prestataire ou cliente).

Cela exigera de remettre à plat les processus, en essayant de retrouver comment les faire fonctionner « à la main », pour réimaginer, par étape, leur automatisation de façon incrémentale.

Revaloriser l'homme : un projet utile et mobilisateur ?

Ce travail est aujourd'hui à peine commencé, mais il offre par contre, du point de vue de Bruno Lussato, des perspectives motivantes et même enthousiasmantes : réduire le gaspillage et redonner leur fierté et leur responsabilité aux femmes et aux hommes de terrain.

L'auteur conclut donc en nous engageant, en même temps qu'à une sorte de petite défiance, voire de « résistance » à l'organisation MATRIX, à participer à la construction d'une REFONDATION de l'organisation de l'informatique et du monde de l'entreprise d'aujourd'hui.

C'est sans doute un défi redoutable et audacieux, engagé et responsable. Un challenge qui nous met pleinement seuls, en même temps que collectivement, face à notre responsabilité, certes un peu vertigineuse, de « citoyens ». À chacune et à chacun d'entre nous d'apprécier si ce défi est digne ou non de d'être relevé...

Pierre Fischhof
Animateur de la commission Adeli «Informatique et être humain»
Chef de projet en refontes et migrations de systèmes d'information
Pierre.fischhof@libertysurf.fr

Bibliographie :

- François de Closets, Bruno Lussato. *L'imposture informatique*. Fayard, 2000.
- Bruno Lussato, Olivier de Tissot, Philippe Delannoy. *L'échelle humaine*. Robert Laffont, 1999.
- Bruno Lussato. *La troisième révolution*. Plon, 1999.
- Bruno Lussato *Décodage*. Interéditions, 1997
- Bruno Lussato. *Claude Mediavilla : du signe calligraphié à la peinture abstraite*. Bordas, 1996
- Bruno Lussato, Jean-Pierre Gueno *Un amour de stylo*. Robert Laffont, 1995.
- Bruno Lussato. *La théorie de l'empreinte, Eléments pour une théorie de l'information psychologique*, ESF Editeur, Communication et complexité, 1991.
- Bruno Lussato *Le défi culturel*. Nathan, 1991.
- Bruno Lussato, Gérald Messadié. *Bouillon de culture*. Robert Laffont, 1986.
- Bruno Lussato. *Le défi informatique*. Hachette, 1981.
- Bruno Lussato *Les structures de l'entreprise*. Organisation, 1981.
- Bruno Lussato *Introduction critique aux théories d'organisation* Dunod, 1972.