

Titre: Les aspects de l'échange d'information dans un processus de génie logiciel
Title: logiciel

Auteurs: Patrick D'Astous, & Pierre N. Robillard
Authors:

Date: 1997

Type: Rapport / Report

Référence: D'Astous, P., & Robillard, P. N. (1997). Les aspects de l'échange d'information dans un processus de génie logiciel. (Rapport technique n° EPM-RT-97-06).
Citation: <https://publications.polymtl.ca/9713/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/9713/>
PolyPublie URL:

Version: Version officielle de l'éditeur / Published version

Conditions d'utilisation: Tous droits réservés / All rights reserved
Terms of Use:

 **Document publié chez l'éditeur officiel**
Document issued by the official publisher

Institution: École Polytechnique de Montréal

Numéro de rapport: EPM-RT-97-06
Report number:

URL officiel:
Official URL:

Mention légale:
Legal notice:

20 JUIN 1997

ÉCOLE POLYTECHNIQUE
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ET DE GÉNIE INFORMATIQUE

LES ASPECTS DE L'ÉCHANGE D'INFORMATION
DANS UN PROCESSUS DE GÉNIE LOGICIEL

qualité

Patrick d'Astous
Étudiant Ph.D.,

Pierre N. Robillard Ph.D., ing.
Directeur

Laboratoire de recherche en génie logiciel
École Polytechnique de Montréal
C.P. 6079, suc. Centre-ville
Montréal, qc. H3C 3A7
dastous@rgl.polymtl.ca
robillard@rgl.polymtl.ca
Tel: 514-340-4238
Fax: 514-340-3240

Janvier 1997

Tous droits réservés. On ne peut reproduire ni diffuser aucune partie du présent ouvrage, sous quelque forme ou par quelque procédé que ce soit, sans avoir obtenu au préalable l'autorisation écrite des auteurs.

Dépôt légal, 01/97
Bibliothèque nationale du Québec
Bibliothèque nationale du Canada

Les aspects de l'échange d'information dans un processus
de génie logiciel (EPM/RT-97/06)

Patrick d'Astous, Pierre N. Robillard (Génie électrique et génie informatique)

Pour se procurer une copie de ce document, s'adresser au:

Service des Éditions
École Polytechnique de Montréal
Case postale 6079, Succursale Centre-ville
Montréal (Québec) H3C 3A7
Téléphone: (514) 340-4473
Télécopie: (514) 340-3734

Compter 0,10 \$ par page et ajouter 3,00 \$ pour la couverture, les frais de poste et la manutention. Régler en dollars canadiens par chèque ou mandat-poste au nom de l'École Polytechnique de Montréal.

Nous n'honorons que les commandes accompagnées d'un paiement, sauf s'il y a eu entente préalable dans le cas d'établissements d'enseignement, de sociétés ou d'organismes canadiens.

INTRODUCTION	3
PARTIE 1 - ÉTAT DES CONNAISSANCES	8
L'APPROCHE TECHNIQUE AU TRAVAIL D'ÉQUIPE	8
Les principes du travail d'équipe	8
Les équipes à succès	10
L'art du travail d'équipe	11
LA PRISE DE DÉCISION	11
Termes importants	12
Approches à la prise de décision	13
Le modèle en spirale	15
Sommaire de la prise de décision	17
LES OUTILS DE SOLUTION DE PROBLÈMES	17
LE DESIGN DE LOGICIELS EN GROUPE	19
DES EQUIPES DE TRAVAIL EFFICACES	22
Le modèle de diagnostique du travail	23
L'indicateur de type Myers-Briggs	23
Le modèle des rôles de Belbin	25
LE DESIGN OO ET LES GROUPES DE TRAVAIL	27
Aspects organisationnels	28
SOMMAIRE	28
PARTIE 2 - PROTOCOLE DE RECHERCHE	31
CONCEPTUALISATION DU PROBLÈME DE RECHERCHE	31
Objectif général de la recherche	31
Cible de la recherche	31
Importance du sujet de la recherche	31
Formulation du problème de recherche	32

Stade de développement de la recherche	33
Utilisation des résultats de la recherche	33
Modèle théorique retenu	34
Hypothèse de recherche	35
STRATÉGIE DE RECHERCHE	35
Approche générale	35
Validité de la stratégie de recherche	36
Population cible	37
Échantillon	37
Variable dépendante	37
Variable indépendante	38
Méthode de collecte des données	38
Qualité des instruments de mesure	39
Validité de la mesure	39
Analyse des données	40
SOMMAIRE	40
RÉSULTATS INITIAUX OBTENUS	42
TRAVAUX FUTURS	44
COLLABORATION	44
CONCLUSION	45
RÉFÉRENCES	47

INTRODUCTION

Au cours de la dernière décennie, le logiciel est devenu tranquillement mais sûrement l'une sinon la plus importante partie de la plupart des systèmes fabriqués. En fait, les logiciels pilotent maintenant nos avions [1], contrôlent les débits automatiques [2] et même les moteurs de nos automobiles [3].

La conception de logiciels est donc une facette importante du développement de nouveaux systèmes. Pourtant, plusieurs projets logiciels ne se concrétisent jamais ou produisent un produit qui ne satisfait pas le client. Une étude faite auprès du Département de la Défense (DOD) américain a démontré que moins de 2 % des logiciels acquis étaient utilisables dès leur achat [4]. L'application des principes et philosophies du génie au développement du logiciel a réduit considérablement le problème, mais dix ans plus tard, il reste encore beaucoup de travail à faire.

Selon Schach [5], le but du génie logiciel est de produire du logiciel de qualité, et ce dans les délais et les budgets prévus. Cette définition requiert donc des ingénieurs logiciels non seulement des connaissances techniques exhaustives mais aussi la connaissance de notions de gestion de projets. La gestion de projet est définie comme l'art de diriger et coordonner les ressources humaines et matérielles d'un projet dans le but de réaliser des objectifs prédéterminés de spécifications, temps, budget et qualité [6]. Ces deux définitions sont non seulement similaires, mais elles permettent aussi de rapprocher deux champs d'expertise complémentaires.

Les processus de création de génie logiciel sont relativement nouveaux, ayant évolué à partir de simples façons de faire pour devenir des modèles reconnus. On peut affirmer que 1985 fut l'année où vraiment le concept de processus prit naissance. Dès les premières années, on se rendit compte que la pratique du génie logiciel faisait face à des difficultés inhérentes et propres qui sont encore aujourd'hui incontournables. Le Professeur Brooks [7], dans son article intitulé "No silver bullet" divisa ces difficultés en deux catégories: les essences et les accidents.

Les essences représentent ces difficultés qui sont inhérentes à la création de logiciels et qui doivent être gérées pour obtenir un produit satisfaisant. La complexité est inhérente à un logiciel parce qu'aucune partie n'est identique à une autre. Ce problème n'est pas seulement technique mais demande aussi une gestion complète du logiciel afin d'y contrôler son intégrité. La conformité est nécessaire afin d'intégrer le logiciel à d'autres parties d'un système. Un logiciel doit être versatile parce qu'il est toujours nécessaire de le modifier afin de répondre aux besoins techniques de la plate-forme ou aux besoins des utilisateurs. Le logiciel est une partie d'un système et est de ce fait influencé par les autres parties de ce système. Enfin, il est difficile de visualiser les structures d'un logiciel puisqu'en fait, elles n'existent pas dans la réalité.

C'est ce qui fait du développement de logiciels une discipline de génie unique en son genre. Le produit, le logiciel, est une entité abstraite qui n'existe qu'avec le support de médias ou de n'importe quel engin compatible. Le processus de conception demande donc au développeur une imagination lui permettant de voir des structures ou des caractéristiques d'un produit abstrait. Contrairement à d'autres disciplines du génie qui permettent au concepteur de se baser sur des techniques prouvées et des données empiriques lui permettant de créer un design en utilisant en grande partie son expérience, chaque logiciel est unique car c'est le logiciel qui habituellement doit se plier aux exigences de son environnement.

Bien sûr, un programmeur doit se plier à des règles et des techniques pour accomplir son travail, mais en grande partie, il doit se fier à son imagination pour se sortir d'embûches tout au long de la conception d'un logiciel. L'aspect cognitif du processus est donc très important parce qu'elle permet d'identifier ce que l'humain doit comprendre. Cette notion place donc une vaste portion du risque non pas sur les caractéristiques techniques du produit mais sur les aptitudes des participants au processus à pouvoir comprendre et régler les problèmes techniques.

Un projet de conception de logiciel est, par définition, une entreprise qui permet la création de quelque chose de nouveau dont il n'existe aucune réplique disponible

dans son ensemble. C'est à cette situation que répond le génie logiciel par la création de processus permettant la production d'un logiciel par une suite d'étapes logiques. Une équipe est formée, souvent composée de ces "champions de la programmation", experts dans leur domaine mais n'ayant jamais travaillé en équipe dans la poursuite d'un but commun. L'ingénieur logiciel responsable du projet doit être capable de comprendre les principes fondamentaux de sa discipline et il doit aussi devenir une personne capable de comprendre le travail d'équipe et appliquer des principes de gestion du personnel et de leadership afin que le projet soit un succès.

Le paradigme de Van Scoy [8] mentionne que la communication est l'élément essentiel de la gestion du processus. Cette affirmation ne fait que toucher une infime partie des difficultés auxquelles on fait face. Que penser de la structure de l'organisation elle-même, permet-elle aux participants d'utiliser leurs aptitudes au plus fort de leur potentiel ? L'adage disant que deux têtes valent mieux qu'une s'applique sûrement au design de logiciels, alors pourquoi est-ce que plusieurs programmeurs n'aiment pas travailler en équipe ?

La communication est l'un des plus importants éléments permettant d'harmoniser les connaissances dans un groupe et de s'assurer que les besoins technologiques sont bien compris. La communication est souvent difficile parce que les caractéristiques du logiciel et les problèmes associés avec son développement sont mal compris. Chaque individu approche le problème "d'écrire un programme" avec son propre bagage de connaissances et d'expérience. La qualité d'un logiciel est donc intimement liée à la façon de communiquer l'information et l'expertise nécessaire à l'accomplissement du travail à l'intérieur du groupe.

L'aspect organisationnel du génie logiciel a donc une grande importance car c'est un fait indiscutable que la difficulté de trouver une solution à un problème n'est habituellement pas la conséquence d'une insuffisance de la technologie mais plutôt d'une mauvaise application de celle-ci. En fait, le logiciel n'est pas manufacturé au sens propre du mot, il est plutôt développé et conçu. Une bonne qualité est donc reflétée par un bon

design. Le processus est donc très dépendant des personnes. Il existe plusieurs types d'instruments de communication permettant de disséminer l'information à l'intérieur d'un groupe. L'évaluation de ces instruments dans le processus de génie logiciel est donc une recherche importante et nécessaire.

Plus qu'aucune autre forme de création technique, le génie logiciel demande de chaque individu des aptitudes à pouvoir imaginer et créer des entités abstraites et surtout à être capable de relier ces entités ensemble grâce à des liens qui eux aussi sont abstraits. Ce sont ces caractéristiques qui rendent le travail en équipe très difficile car il n'existe pas deux personnes qui imaginent la même solution à un problème très précis. C'est pour cette raison que beaucoup d'informaticiens et d'ingénieurs en informatique appréhendent le moment où ils devront concevoir un logiciel en compagnie d'autres personnes. Bien sûr, le génie logiciel vient en aide à ces gens en leur donnant des processus et des outils de mesures qui permettent de mieux comprendre leur rôle dans le développement d'un logiciel.

La pression exercée sur ces concepteurs par les contraintes de budget et les échéanciers rend presque impossible la création artisanale de logiciel, c'est-à-dire que chaque logiciel soit créé à partir de rien et où le succès dépend des idées et de l'expertise de seulement quelques personnes. Il faut donc travailler en équipe dans le but de diriger toutes les ressources techniques et l'expertise vers un but commun: le développement d'un logiciel.

L'idée de former des équipes pour concevoir et implanter un logiciel n'a rien de nouveau. Elle n'a cependant pas toujours eu les résultats escomptés et a souvent été critiquée par les informaticiens qui trouvent que le processus diminue leur créativité. La performance d'une équipe et sa productivité est proportionnelle à la communication à l'intérieur du groupe autant qu'avec le monde extérieur. Il faut donc évaluer les méthodes de communication afin de déterminer leur efficacité à permettre le développement de produits de qualité.

La communication informelle fait partie intégrale de tout groupe. En effet, les

conversations régulières entre les membres de l'équipe sont aussi des outils importants de transfert d'information et d'idées. Ce transfert sera de plus en plus efficace au fur et à mesure que le groupe se soudera ensemble. La communication informelle occupe donc une place importante dans le processus de développement de logiciels.

Les communications dites formelles sont effectuées grâce à des réunions durant lesquelles les membres de l'équipe peuvent échanger leurs idées, évaluer le travail accompli ou décider des tâches à accomplir. Il existe deux grands types de réunions utilisées dans le processus de génie logiciel: les réunions de travail et les revues. Les réunions de travail permettent aux spécialistes d'établir les spécifications du produit et d'échanger les mérites de solutions envisagées de façon individuelle dans le but de déterminer la meilleure marche à suivre ou la meilleure solution aux problèmes. Les revues sont des réunions qui apparaissent tout au long du processus de développement et qui servent à s'assurer que les spécifications du produit répondent aux besoins. Les participants aux revues parcourent le document afin de s'assurer qu'il n'existe aucune ambiguïté quant à son contenu [5].

Ces réunions de toute sorte sont donc très importantes pour le bon fonctionnement d'un projet et la production d'un produit de qualité. Elles occupent en fait une proportion importante du temps consacré à un projet et doivent générer des résultats satisfaisants. Il est donc nécessaire de savoir quels éléments sont importants dans ces réunions et lesquels font qu'une réunion est plus productive qu'une autre.

PARTIE 1 - ÉTAT DES CONNAISSANCES

L'APPROCHE TECHNIQUE AU TRAVAIL D'ÉQUIPE

Les principes du travail d'équipe

Les ingénieurs sont au courant qu'il existe habituellement une suite logique d'actions qui doivent être prises lorsqu'un processus est complexe et qu'une action dépend du

résultat de l'action précédente. D'après Berger [9], le travail d'équipe demande lui aussi plusieurs tâches qui doivent être accomplies dans chacune de ses phases:

Préparation au travail. La préparation au travail est une étape où les membres du groupe tentent de se trouver une place à l'intérieur de l'équipe. Le chef d'équipe doit donc s'assurer que tous les membres ont la chance de se connaître, de démontrer leurs habiletés distinctives, de clarifier leurs attentes et les règles qui régiront l'équipe. Ils doivent aussi se mettre d'accord sur la mission et les objectifs de l'équipe.

Défi à l'autorité. Après que les équipiers aient compris leur rôle et la mission de l'équipe, plusieurs coalitions se formeront à partir de personnes ayant le même point de vue et une lutte s'ensuivra afin de déterminer qui contrôlera l'équipe et comment elle le sera. Le point de mire de ce défi à l'autorité sera en premier lieu le chef d'équipe mais cela ne tardera pas à s'étendre aux autres leader potentiels. Les tâches de leadership nécessaires sont: une réponse équitable et calme aux défis des membres, la renégociation des attentes, la médiation entre les coalitions et la répartition du travail selon la préférence des membres et les ressources disponibles.

Production. Lorsque les membres de l'équipe se sentent assez confortables et qu'ils croient que tous les mécanismes sont en place pour partager le travail et résoudre les dissensions de façon équitable, l'équipe entre dans la phase de production où la qualité et la quantité du travail augmente radicalement. Le travail ne fait plus l'objet de distractions causées par des discussions sur le processus qui gaspillent temps et énergies. Le chef d'équipe doit continuer de contrôler et supporter les attentes et les standards de l'équipe, explorer de nouvelles façons d'appuyer la performance des individus et de faciliter la résolution de problèmes.

Rapprochement. Les équipes productives ayant du succès ont souvent éprouvé des sentiments de joie et de camaraderie. La cohésion est issue d'une intégration délicate de la recherche individuelle de l'autonomie et de la participation réussie dans un groupe. Le chef d'équipe doit donc fournir des chances de célébrer les bons coups, mettre en

évidence les contributions des différents membres de l'équipe, partager la reconnaissance et les récompenses dues à l'équipe et démontrer sa satisfaction pour un travail bien fait.

Les équipes à succès

Voici donc une liste des caractéristiques essentielles d'une équipe ayant du succès:

Leadership consistant et approprié. Un leadership doit s'ajuster à la situation. Un style participatif n'est peut-être pas approprié dans tous les cas. Un bon leader doit être capable d'exercer un leadership qui permettra à son équipe d'atteindre son but.

Objectifs clairs et concis. Il est important d'obtenir une vision claire, bien comprise et acceptée des objectifs. Une équipe qui a du succès doit permettre à ses membres de discuter, clarifier et négocier les objectifs de l'équipe. De cette façon, chaque membre connaît les priorités et comment y arriver.

Intégration des buts personnels et collectifs. Les membres d'un équipe ne s'engagent pas envers les objectifs d'équipe automatiquement. Chaque membre doit parvenir à des objectifs personnels à l'intérieur de l'équipe. Leur motivation sera donc proportionnelle à leur possibilité de se développer et de grandir tout en travaillant.

Règles équitables et consistantes. Les règles qui gouvernent les interactions doivent être équitables, appliquées constamment et sujets au changement si nécessaire. Les équipes à succès s'adaptent aux conditions en ajoutant de nouvelles règles et en enlevant celles qui sont désuètes.

Équipiers qualifiés et diversifiés. Les équipes performantes semblent avoir chacun de ses postes remplis. Le mélange d'habiletés et d'idées permet à l'équipe d'utiliser les bonnes ressources pour un problème particulier. Malgré que ce mélange amène des conflits, une bonne équipe parvient à gérer ces différences et à en tirer des avantages.

Climat approprié. Le climat de travail est aussi important que le processus de travail de

l'équipe. Une bonne équipe prend soin de ses membres en créant un climat de respect, d'excellence, d'innovation et de volonté d'aller toujours plus loin.

Engagement. Une caractéristique commune des équipes à succès semble être un engagement total. Les équipes travaillent fort en s'occupant des détails, en anticipant les problèmes et en s'organisant. Lorsque le temps de relaxer survient, les activités de loisirs consolident les liens entre les membres et réduisent le stress du travail.

Le processus de travail d'équipe ne peut être réduit complètement en principes et techniques sans y ajouter un autre concept, l'art du travail d'équipe.

L'art du travail d'équipe

Attendre le bon moment. Un chef d'équipe doit être capable de déterminer le moment propice pour introduire une nouvelle règle, identifier un nouveau problème ou proposer une nouvelle solution. Une bonne compréhension du rythme naturel des événements afin d'attendre le dernier moment avant d'intervenir augmentera la probabilité que cette intervention soit profitable.

Association avec les autres membres. Les leaders ne doivent pas ressortir du groupe mais plutôt tenter de se confondre en supportant les autres dans l'accomplissement des objectifs de l'équipe.

Entrevoir les possibilités. Le chef d'équipe doit être capable d'évaluer de nouvelles façons d'atteindre ses objectifs. Il doit avoir une connaissance de la situation, une bonne appréciation des possibilités et la volonté de s'éloigner des conventions pour accomplir le travail.

LA PRISE DE DÉCISION

La plupart des réunions faisant parties du processus de développement logiciel ont pour but, qu'il soit explicite ou implicite, de prendre une décision afin de permettre au projet de continuer. Cette décision peut être de nature technique, de gestion ou de tout

autre type mais elle affecte le déroulement du projet. Il est donc important, dans un premier lieu, de comprendre le processus qui régit la prise de décision en groupe. Il faut cependant stipuler qu'il n'existe aucune "formule" qui permette d'assurer que la décision prise par un groupe sera la bonne mais une bonne connaissance de ce processus ne peut qu'aider à sensibiliser les participants.

Termes importants

Cette section fera le survol des termes importants utilisés dans le processus de prise de décisions en groupe. Tous ces termes sont familiers et peuvent être immédiatement définis de façon globale, mais il est nécessaire de comprendre précisément ce qu'ils veulent dire.

La relation entre la *prise de décision* et la *solution à un problème* n'est pas universellement reconnue. Plusieurs voient ces deux expressions comme des synonymes tandis que d'autres y voient deux significations tout à fait différentes. Il existe deux types de problèmes: ceux pour lesquels il existe une solution correcte habituellement déterminée par des moyens externes ou objectifs et ceux dont il n'existe aucune solution pouvant être validée de façon externe. Ces solutions ne sont découvertes que par l'acceptation du groupe et l'engagement des individus à implanter cette solution.

Une *décision* est l'ultime résultat de l'interaction de groupe. Elle est inévitablement un choix, fait par les membres du groupe parmi un nombre d'alternatives disponibles. Malgré que la quantité d'alternatives ne soit pas une mesure de la productivité d'un groupe, il ne faut pas oublier que seulement les alternatives proposées durant l'interaction du groupe seront disponibles lors de la prise de décision finale. D'ailleurs, le processus de décision n'est qu'une suite d'essais, de la part de chacun des membres, à influencer directement ou indirectement les autres membres à accepter ou rejeter une alternative proposée.

Les membres d'un groupe parviennent à une décision lorsqu'il y a *consensus*. Une décision prise par consensus demande donc l'accord de tous les membres, mais

cela n'est pas une condition suffisante au consensus. En effet, le consensus implique l'engagement envers la décision prise. L'élément essentiel du consensus est donc l'importance de la loyauté au groupe partagée par ses membres.

Approches à la prise de décision

Il existe deux approches distinctes mais très utilisées au processus de prise de décision en groupe. Chacune ayant ses limitations et ses avantages. L'approche *prescrite* tente d'expliquer comment les groupes devraient prendre les décisions tandis l'approche *descriptive* documente la façon dont les groupes prennent leurs décisions.

Les approches prescrites sont basées sur quelques assomptions. Premièrement, les méthodes prescrites assument que tous les membres du groupe sont constamment rationnels. Ces méthodes fournissent habituellement une liste des étapes que le groupe doit franchir pour parvenir à une décision. De plus, elles préviennent contre les émotions et les aspects non-rationnels de l'interaction du groupe qui peuvent déranger l'efficacité du processus de décision. Une seconde assomption est que l'utilisation de ces méthodes mènera invariablement vers un groupe plus productif et que cela améliorera la qualité des résultats du groupe de décision.

Le plus connu des modèles prescrits est sûrement la "pensée réflexive" suggérée par John Dewey [10]. Malgré que Dewey ait bâti son modèle en fonction du processus mental d'un individu seulement, il a été beaucoup utilisé comme guide à la décision de groupe. Le modèle de Dewey inclut les six étapes suivantes:

1. Une difficulté est ressentie ou exprimée.
2. La nature du problème est définie.
3. La nature du problème est analysée.
4. Les solutions possibles sont suggérées pour résoudre le problème.
5. Les solutions sont comparées en les évaluant par rapport à des critères présélectionnés et la meilleure solution est choisie.
6. La meilleure solution est implantée.

Ce modèle sert donc d'agenda universel qui guide un groupe vers un consensus dans une progression étapes par étapes. Le groupe doit discuter et compléter chaque étape avant de passer à la suivante. Lorsque ce modèle est utilisé pour guider la prise de décision en groupe, il ne permet pas l'aspect socio-émotionnelle de s'ingérer dans le processus.

Les approches descriptives sont basées sur le fait qu'il existe déjà un processus "naturel" de décision dans un groupe. Ce processus est présent en fonction de la liberté que le groupe possède à développer ses propres tâches et ses propres dimensions sociales sans l'influence d'aucune source externe.

Le modèle le plus familier de l'approche descriptive est le modèle en trois phases avancé par Bales [11]. Après l'observation de plusieurs groupes de décision, il a été signalé que les membres de ces groupes tendaient à discuter différentes sortes de problèmes ayant trait à leurs tâches à différentes périodes de l'interaction du groupe. A partir des données compilées grâce au système d'analyse du processus d'interaction (IPA) de Bales, trois phases ont été identifiées:

1. Emphase sur les problèmes d'orientation (décision sur l'évaluation de la situation actuelle).
2. Emphase sur les problèmes d'évaluation (décision sur l'attitude à prendre vis-à-vis cette situation).
3. Emphase sur les problèmes de contrôle (décision sur les actions à prendre).

Le IPA classe chaque communication durant l'interaction du groupe selon douze catégories différentes qui sont en fait six paires de catégories bipolaires:

CATÉGORIE	CATÉGORIE	PROBLÈME
Est solidaire	Est antagoniste	Problème d'intégration
Montre une baisse de tension	Montre un hausse de tension	Problème de tension
Est en accord	Est en désaccord	Problème de décision
Donne des suggestions	Demande des suggestions	Problème de contrôle

Donne son opinion	Demande un opinion	Problème d'évaluation
Donne de l'orientation	Demande de l'orientation	Problème d'orientation

Figure 1 Le modèle de Bales

Chaque paire de catégories est associée à un problème particulier auquel le groupe est confronté. L'emphase est portée sur les problèmes d'orientation, d'évaluation et de contrôle par l'utilisation de trois phases successives durant le processus. Il semble que ces trois phases soient cycliques puisque lorsqu'un groupe complète un tâche de prise de décision en utilisant les trois phases, le groupe retourne à la phase d'orientation au début de la tâche suivante.

Afin d'être consistant avec la dimension socio-émotionnelle présente durant le processus, il est requis que la structure sociale soit stable pour obtenir de la productivité. Durant les phases médianes, le modèle reflète une période de conflit entre les membres et de différences d'opinion sur les tâches et les normes sociales.

Le modèle en spirale

Bien que les deux approches décrites dans la section précédente soient différentes, elles ont en commun le fait qu'elles soient linéaires. Chacun des modèles est basé sur une progression étapes par étapes vers le résultat escompté. Scheidel et Crowell [12] ont découvert que le processus d'interaction dans la prise de décision n'était pas linéaire. Le modèle en spirale a donc été développé pour représenter ce phénomène.

Lorsqu'un membre du groupe introduit une nouvelle idée, les autres membres sont en accord ou en désaccord ou proposent des modifications. Cette nouvelle idée est l'objet de la discussion et elle est développée pour refléter le point de vue du groupe jusqu'au point où tous les membres du groupe sont en accord. Le groupe ancre donc sa position et introduit une nouvelle idée préliminaire à partir de l'idée précédente. Ce processus en spirale se sert donc de ces points d'ancrages comme base pour la prochaine discussion sur l'idée suivante. Si un consensus est établi pour cette nouvelle idée, un nouveau

point d'ancrage est établi. Si l'idée est rejetée, alors le groupe retourne au dernier point d'ancrage pour repartir avec une nouvelle idée.

Le processus en spirale est cumulatif et progressif. Il permet de refléter les modifications continues des idées et le retour vers les consensus établis pour reconfirmer la position du groupe. De plus, ce modèle prend en compte l'inefficacité apparente du groupe de décision par rapport à l'utilisation du temps. De plus, il est raisonnable de conclure qu'un groupe ne prend pas des décisions mais plutôt que les décisions émergent de l'interaction du groupe.

En observant les groupes de prise de décisions, B. Aubrey Fisher [13] a découvert quatre phases au processus d'émergence des décisions. Contrairement à d'autres types d'observations, chacune des alternatives de décisions ont été identifiées et le processus d'interactions durant la conversion de ces idées préliminaires en un consensus a été surveillé. Les données ainsi recueillies ont permis d'établir quatre phases distinctes dans la décision de groupe, chacune d'elles caractérisées par un type d'interaction très différent.

La phase d'*orientation* est une période servant à former des opinions et à se débarrasser des inhibiteurs sociaux. Elle sert à se connaître, à clarifier et à tenter d'exprimer des attitudes envers les autres. La phase de *conflit* est caractérisée par la dispute et les conflits idéologiques vis-à-vis les propositions alternatives.

La phase d'*émergence* est probablement l'étape cruciale du processus de décision de groupe. Durant cette phase, le résultat possible de l'interaction de groupe devient de plus en plus visible. La phase de *renforcement* sert à développer l'engagement des membres envers les décisions qui étaient sujets à conflit durant la seconde phase et qui ont émergé durant la troisième phase.

Sommaire de la prise de décision

Le processus de prise de décision englobe la recherche d'une solution à un problème, ce qui demande un taux élevé d'acceptation de la solution. Une décision constitue un choix entre différentes solutions dont la somme peut être considérée comme la performance du groupe.

Le processus de prise de décisions en groupe est comparé au processus de leadership et de l'émergence des rôles afin de démontrer que les décisions ne sont pas prises par un groupe mais plutôt qu'elles émergent de l'interaction du groupe.

Un groupe parvient à un consensus sur une décision à prendre en modifiant une suite de propositions dans un processus où les alternatives sont introduites, discutées, écartées et réintroduites avec des petites modifications jusqu'au moment où un consensus est établi.

LES OUTILS DE SOLUTION DE PROBLÈMES

Bien que le processus de prise de décisions en groupe soit connu et que cette notion soit claire dans l'esprit de tous les membres du groupe, il se peut que l'on se rende compte qu'il manque toujours quelque chose pour assurer le succès de la réunion. Il est donc important que la personne agissant comme facilitateur soit au courant de la dynamique présente lors de la solution de problèmes en groupe et qu'elle ait une bonne connaissance des nombreux outils de prise de décisions disponibles.

La première étape dans la solution de problèmes en groupe est de s'assurer que le problème est perçu de la même façon par tous les membres [14]. Le meilleur moyen d'établir une définition commune est de demander à chacun des participants son point de vue quant à la nature du problème. L'étape suivante est logiquement d'obtenir une définition précise du problème dans des termes pouvant être utilisés lors des étapes futures. De plus, la définition du problème limite le nombre d'alternatives possibles,

appelées l'étendue du problème, qui est elle-même proportionnellement relative au nombre de solutions possibles.

Durant la phase d'analyse, l'objectif est de partager le problème en parties et d'examiner comment elles interagissent entre elles. Tous les problèmes peuvent être subdivisés récursivement en plus petits sous problèmes jusqu'au point où ils sont simples à comprendre. Il est aussi important de développer une liste de solutions existantes à un problème et d'en établir une liste des avantages et des désavantages.

Il est important de constater qu'il existe plusieurs méthodes qui peuvent être utilisées dans différentes phases du processus. Par exemple, le *brainstorming* peut servir à générer des problèmes, des définitions, des critères et même des alternatives. Le brainstorming est l'une des plus simples et des plus efficaces méthodes pour générer des réponses aux questions du groupe. Les règles sont simples: tous et chacun lancent les idées qui leur passent par la tête afin qu'elles soient prises en note et aucun autre membre n'a le droit de critiquer l'idée avant que la session de brainstorming ne soit terminée.

Une autre méthode plus systématique d'analyser un problème et de générer des alternatives est appelée la méthode de l'échiquier ou l'analyse morphologique. Cette technique oblige les participants à examiner les différentes interactions et les combinaisons de deux ensembles de variables. Les carrés formés à partir des intersections de deux variables sont considérés comme des alternatives distinctes. Cette technique permet de trouver un plus grand nombre de possibilités et d'examiner le problème sous plusieurs angles.

La clé pour évaluer les alternatives en groupe est d'établir les critères de sélection avant de juger celles-ci. La présélection des critères permet d'éliminer la sélection d'alternatives à partir de critères personnels conscients ou subconscients. Cette étape importante est souvent omise lors du processus de solution de problèmes ou de prise de décisions.

Lors de la phase de prise de décision, le groupe devrait revoir les résultats de l'évaluation et s'engager fermement envers un ou plusieurs champs d'actions. Les méthodes d'interaction de réunions de groupes sont établies en fonction de produire une situation où tous les partis y trouvent leur compte.

La rétroaction est le moyen utilisé pour examiner les résultats du processus de solutions de problèmes. L'information ainsi reçue permet de valider le processus et d'y apporter les correctifs nécessaires.

LE DESIGN DE LOGICIELS EN GROUPE

Le domaine du génie logiciel est de plus en plus attentif aux besoins des individus pratiquant la discipline afin d'avoir les meilleurs outils disponibles. C'est dans cet optique que l'étude de la collaboration entre les individus devient de plus en plus importante. Pourtant il n'existe que très peu de littérature sur la façon dont le design de logiciels est fait.

La plupart des systèmes qui sont produits en équipe le sont à cause de leur complexité, le besoin d'une expertise très spécialisée ou d'un échéancier très serré. Le design de logiciels est un exemple classique d'un problème mal structuré [15] et est donc considéré comme une tâche difficile, même pour une seule personne. La coordination de ces activités à l'intérieur d'un groupe est donc passablement difficile et les réunions sont donc très importantes pour coordonner les activités des membres de l'équipe.

Olson [16] a observé des groupes d'ingénieurs logiciels discutant les premières ébauches de systèmes qui seraient produits plus tard, habituellement par d'autres. Ces groupes avaient reçu une description générale du système en question et leurs discussions furent un mélange de spécifications des besoins et de design de haut niveau. En termes de processus logiciel, ces réunions faisaient parties des étapes de spécifications et de design. Toutes ces réunions comprenaient un groupe de personnes assis autour d'une table avec pour seuls outils du papier, un crayon et un tableau.

L'analyse des discussions a permis de développer un ensemble de catégories d'activités qui reflètent la nature des discussions:

Sujet: Les questions et aspects majeurs concernant l'objectif du projet. L'emphase est mis sur des sujets tels que "Devrait-on offrir cette fonctionnalité à l'utilisateur ?" ou aussi "Comment peut-on implanter cette fonction ?". Cela inclut aussi l'élaboration d'une idée ou d'une description qui n'est pas une réponse à une question d'un membre du groupe.

Parfois, les sujets ne seront pas indiqués explicitement mais plutôt inférés par la présentation de deux alternatives. Ceux-ci marqueront typiquement la première partie d'une phrase qui présentent ces alternatives.

Alternatives: Solutions ou propositions concernant l'objectif du projet. Ce sont des caractéristiques pouvant être offertes à l'utilisateur ou des façons d'implanter des caractéristiques préalablement choisies. Cela inclut l'élaboration d'une idée dont la description n'est pas une réponse à une question d'un membre du groupe. Parfois, cette catégorie inclura l'élaboration des implications de l'idée présentée.

Critères: Les raisons, arguments ou opinions qui évaluent une solution alternative ou une proposition. Les critères apparaîtront parfois sous la forme de systèmes analogues en supposant que si une solution fonctionne avec l'autre système, elle devrait le faire avec le système actuel. Parfois, une évaluation ou un critère seront dits sans connexion avec aucune alternative dans le but que les membres du groupe soient rappelés d'évaluer les solutions futures ou passées pour en tenir compte.

Gestion de projet: Déclarations concernant des activités indirectement reliées au contenu du design, par lesquelles les individus se font assigner certaines tâches, décident de la prochaine rencontre, font un rapport sur un activité (ne contenant aucun design) assignées à la réunion précédente, etc.

Gestion de la réunion: Déclarations à propos de la gestion de la réunion actuelle, telles que brainstorming, décisions, votes, arrêts des discussions, etc.

Sommaire: Revues de l'état du design ou de l'implantation, des sujets, alternatives et critères. Le sommaire doit être une déclaration simple (liste).

Clarification: Questions et réponses que quelqu'un demande ou semble ne pas comprendre. Cela inclut les répétitions nécessaires à la clarification, l'association et des explications. Les clarifications servent à éliminer les mauvaises interprétations des autres individus.

Cette catégorie est divisée en dix catégories plus spécifiques et deux nouvelles.

Outils de clarification: Un participant utilise un dessin, une liste ou autre pour clarifier son explication.

Clarification générale: Clarifications qui n'ont pu être associées à une catégorie spécifique.

Interruption: Arrêts des discussions à cause d'une plaisanterie ou une interruption causée par des éléments externes à la réunion. Le changement de cassettes de la caméra devrait être considéré comme une interruption.

- But:** Déclaration de la raison et l'objectif de la réunion du groupe. Les contraintes de travail tels que l'échéancier et la motivation de l'importance de ces contraintes.
- Walkthrough:** Une collection du design fait jusqu'à présent ou les étapes nécessaires à l'utilisateur pour utiliser le design afin de le réviser ou clarifier la situation. Cela correspond habituellement à l'identification des besoins de l'utilisateur, le flux des données ou messages dans l'architecture du système.
- Autre:** Temps utilisé dans la réunion qui ne peut être identifié à une catégorie en particulier.

Après examen des discussions, il a été démontré que les membres de ces réunions mentionnaient clairement les sujets, les alternatives et les critères d'évaluation de ces alternatives en plus de partager leur expérience. Ces catégories ne sont cependant pas complètes et il existe une multitude d'autres choses à expliquer à propos de telles réunions. De plus, cette étude n'a pas tenu compte de la participation des membres et des rôles joués par ceux-ci dans les discussions. Il est certain qu'une meilleure connaissance de ces deux dimensions ne pourraient qu'aider les personnes ayant pour tâche de faciliter ces réunions.

DES EQUIPES DE TRAVAIL EFFICACES

Il est indiscutable qu'une meilleure connaissance de la façon dont une équipe est formée, structurée et menée ne peut qu'améliorer la productivité et que ceci ne peut être atteint sans une évaluation plus rigoureuse des individus formant ces équipes. Il est aussi important de mentionner que les professionnels de l'informatique, forment un bassin de personnes significativement différent des autres groupes habituellement présents à l'intérieur d'une organisation. Thomsett [17] a adapté trois modèles de personnalité, motivation et de leadership bien connus pour des entreprises australiennes ayant plus de

200 spécialistes en informatique. Il a pu enregistrer des augmentations immédiates de productivité de l'ordre de 200 pourcent.

Le modèle de diagnostique du travail

Ce modèle est basé sur les modèles de motivation contemporaine de Emery et Emery [18], Hackman et Oldham [19] et autres. Ce modèle propose que la motivation est intrinsèque et qu'un déterminant majeur de la motivation est la nature et la structure du travail d'une personne. Chaque travail peut être évalué en fonction de cinq dimensions:

1. Le nombre d'habiletés et de talents demandés par le travail d'un individu.
2. La mesure à laquelle l'emploi nécessite l'accomplissement de tâches ayant un résultat visible.
3. La mesure de l'impact du travail accompli sur la vie d'autres individus.
4. Le degré de liberté et d'indépendance de l'individu à établir lui-même ses échéances et ses méthodes de travail.
5. La mesure jusqu'à quel point les activités de son emploi lui permette d'avoir une idée claire de son efficacité et de sa performance.

Un questionnaire permet aux individus d'évaluer leur travail en fonction des cinq dimensions selon une échelle de 1 à 7. Le total est appelé le potentiel de motivation (MPS). Il semble que le MPS d'un travail soit directement relié à la motivation, la productivité, la qualité du travail et autres. En améliorant le MPS des emplois, les équipes pourront ainsi améliorer leur motivation et leur morale.

L'indicateur de type Myers-Briggs

En approfondissant le travail de Carl Jung (1921), Briggs et Myers [20] stipulent que pareillement au fait qu'une personne soit née avec la prédisposition d'être droitier ou gaucher, elle est aussi prédisposée à être extravertie ou introvertie (E ou I), intuitive ou sensée (N ou S), réfléchie ou sensible (T ou F) et juge ou perceptive (J ou P).

Les extravertis sont orientés vers le monde extérieur tandis que les introvertis tendent à se concentrer sur les idées et les sentiments. Les personnes sensées sont concernées par les détails et les faits tandis que les intuitifs préfèrent les possibilités, le futur et l'ensemble. Le type réfléchi prend des décisions basées sur la logique et les critères objectifs pendant que le type sensible prend des décisions à partir de données plus subjectives comme les sentiments et les émotions. Les perceptifs sont flexibles, ouverts et toujours à la recherche de nouvelles informations tandis que les juges veulent garder les choses sous leur contrôle. Les théoriciens ont développé une matrice contenant seize personnalités reliées aux quatre dimensions (Figure 1). Ce modèle a été utilisé par plusieurs chercheurs [21] [22] [23] dans leurs recherches sur les spécialistes en informatique.

		SENSÉE (S)		INTUITIVE (N)	
		RÉFLÉCHIE (T)	SENSIBLE (F)	RÉFLÉCHIE (T)	SENSIBLE (F)
INTROVERTIS (I)	JUGE (J)	ISTJ Sérieux, silencieux, il a du succès grâce à la concentration et la minutie. Il est pratique, ordonné, terre à terre, logique, réaliste et sûr. Il prend des responsabilités.	ISFJ Silencieux, amical, responsable et consciencieux. Il travaille sans relâche pour remplir les obligations. Il est minutieux, assidu et précis. Il est loyal et attentionné.	INFJ Il réussit grâce à sa persévérance, son originalité et son désir de faire tout ce qui est nécessaire. Il est une force silencieuse qui respecte et aide les autres. Il est respecté pour ses principes.	INTJ Il possède de l'originalité et une bonne attitude envers ses propres idées et choix. Il est sceptique, critique, indépendant, déterminé et souvent obstiné.
	PERCEPTIVE (P)	ISTP Spectateur froid, silencieux, réservé et analyste. Il s'intéresse aux principes, comment les choses fonctionnent. Il a des moments d'humour original.	ISFP Retiré, amical, silencieux, sensible et modeste. Il évite les désaccords. Il est souvent relaxe quant aux choses à faire.	INFP Il aime apprendre, les idées, les langages et les projets personnels. Il tend à en faire trop mais il finit par le faire. Amical mais souvent préoccupé.	INTP Silencieux, réservé, impersonnel. Il aime la théorie et les sujets scientifiques. Il s'intéresse aux idées et non aux conversations anodines. Il a des intérêts très particuliers.
EXTROVERTIS (E)	PERCEPTIVE (P)	ESTP Terre à terre, il ne s'inquiète et ne se dépêche pas. Il prend ce qui vient. Il est un peu brusque et insensible. A son meilleur avec les choses qui s'assemblent et se désassemblent.	ESFP Amical et extériorisé. Il rend les choses plus intéressantes pour les autres. Il aime les sports et faire des choses. Il est meilleur à se souvenir des choses qu'à comprendre les théories.	ENFP Enthousiaste, ingénieux et imaginaire. Il est capable de faire tout ce qui l'intéresse. Il est rapide à trouver une solution et à aider les autres qui ont des problèmes.	ENTP Rapide, ingénieux et bon à faire des choses. Il peut argumenter une question pour le plaisir seulement. Très bon pour trouver une solution à un problème difficile, il peut négliger le travail de routine.
	JUGE (J)	ESTJ Pratique, réaliste, terre à terre et un naturel pour les affaires et la mécanique. Ne s'intéresse pas à des sujets s'il n'en voit pas l'utilité. Il aime organiser et contrôler une activité.	ESFJ Chaleureux, populaire consciencieux, coopératif et loquace. Il a besoin d'harmonie. Il est à son mieux quand il est encouragé et il a peu d'intérêt pour l'abstrait et les sujets techniques.	ENFJ Responsable et émotif. Il est sensible à ce que les autres pensent ou veulent. Il est sociable et populaire. Il est sensible à la critique et à l'encouragement.	ENTJ Franc, décisif, un leader. Il est bon avec tout ce qui demande du raisonnement et de l'intelligence. Il est souvent plus optimiste que ce que son expérience lui dicte.

Figure 2 Le modèle Myers-Briggs

Le modèle des rôles de Belbin

Belbin [24] a développé un modèle de rôles à l'intérieur de l'équipe selon lequel le leadership n'émane pas d'une seule personne mais plutôt de chacun des membres de l'équipe à tour de rôle, tout dépendant de la situation ou le problème auquel ils font face. Les membres d'une équipe efficace contribuent en prenant un rôle de leadership ou d'équipe. De plus, il existe un lien important entre le type de personnalité (Myers-Briggs) des membres de l'équipe et le rôle joué dans l'équipe. Belbin a identifié huit rôles d'équipe qui sont nécessaires afin d'obtenir une équipe efficace:

- Président:** Il contrôle la marche de l'équipe vers les objectifs de groupe en utilisant de la meilleure façon possible les ressources disponibles. Il reconnaît où se situent les forces et les faiblesses et s'assure que le potentiel de chacun des membres est utilisé pleinement.
- Façonneur:** Il modèle la façon dont l'effort d'équipe est donné. Il dirige les discussions et cherche à imposer sa façon de faire sur les résultats des activités de groupe.
- Implanteur:** Il amène de nouvelles idées et stratégies en portant attention aux problèmes importants et en se mettant à la recherche de nouvelles approches à un problème auquel le groupe est confronté.
- Moniteur-évaluateur:** Il analyse les problèmes, évalue les idées et les suggestions de façon pratique pour permettre à l'équipe de prendre une décision plus équitable.
- Travailleur:** Il prend les concepts et les plans pour en faire des procédures pratiques. Il met à exécution les plans qui ont été approuvés de façon systématique et efficace.

- Équipier: Il supporte les membres en position de force en ajoutant à leurs suggestions. Il aide ceux qui ont des problèmes. Il améliore la communication entre les membres et rehausse l'esprit d'équipe.
- Investigateur: Il explore et rapporte sur les idées, développements et ressources disponibles à l'extérieur du groupe. Il crée les contacts externes pouvant être utiles à l'équipe et à des négociations ultérieures.
- Minutieux: Il s'assure que l'équipe est protégée contre les erreurs possibles en recherchant activement les aspects du travail qui demandent une attention minutieuse. Il maintient un degré d'urgence dans l'équipe.

Afin de rendre ce modèle plus spécifique au domaine de l'informatique, Thomsett a ajouté un rôle de groupe:

Façonneur technique: Il donne une direction technique au travail soit en analyse, design et programmation. Cette direction est seulement basée sur son expertise technique.

En combinant les trois modèles décrits auparavant, il est possible de créer un outil de diagnostique permettant d'évaluer le rendement des équipes. De plus, ces modèles permettent de restructurer et de former des équipes balancées et efficaces à l'aide de lignes directrices qu'ils rendent disponibles.

Selon le modèle de Belbin, les équipes de programmation types actuelles sont composées d'un programmeur type et d'un chef d'équipe qui partagent le même rôle parce que la nature de la formation en informatique tend à filtrer les autres types de personnalités. Une équipe typique contient donc une concentration de personnalité similaire, ce qui amène de la dissension quant au partage des rôles, un manque de communication, un manque de créativité dans la solution de problèmes et un manque d'esprit d'équipe. Il semble donc qu'un grand nombre d'équipes de développement informatique

manque les habiletés nécessaires à la gestion du personnel et à la négociation si elles sont comparées à d'autres types d'équipes.

Il est donc important de reconnaître la nature spéciale des professionnels de l'informatique afin de créer des équipes efficaces qui permettront des gains de productivité égaux sinon supérieurs à ceux disponibles grâce à l'achat d'outils CASE très dispendieux. L'industrie de l'informatique doit donc utiliser un approche plus systématique à la gestion de son personnel.

LE DESIGN OO ET LES GROUPES DE TRAVAIL

Le design de logiciels complexes est nécessairement un processus social. Il existe cependant plusieurs façons d'organiser les individus pour parvenir à nos fins [25]. La formation d'équipes ou de groupes de travail est la solution employée par les entreprises modernes pour assembler les individus ayant les connaissances et l'expertise nécessaires pour entreprendre un projet particulier. Une équipe doit parvenir à partager ses connaissances d'un problème, de son domaine d'application et du design résultant.

Étant donné la grande difficulté d'analyser et de faire le design d'un logiciel, plusieurs nouvelles méthodes de génie logiciel tentent d'organiser le travail en employant de nouveaux outils et structures. L'une de ces méthodes qui reçoit une grande attention est la méthode orientée-objet. Herbsleb [26] a tenté de comparer les aspects du développement traditionnel avec ceux de la méthode orientée-objet afin d'évaluer les revendications de supériorité de l'orientation objet.

Il en a résulté plusieurs observations intéressantes dont le fait que durant la vie d'un projet donné, le pourcentage de temps passé à travailler en groupe diminuait à mesure que la fin du projet approchait.

Aspects organisationnels

Plusieurs méthodes de design orienté-objet suggèrent qu'un individu ait le rôle de structurer le système et de s'assurer de son intégrité tout au long du processus de développement [27]. Généralement, l'individu qui initie une conversation le plus fréquemment, initiera 40 à 50 pour-cent de toutes les conversations [28]. L'étude de Herbsleb a permis de démontrer que l'architecte-chef d'un projet a été la personne qui a initié le plus de conversations à l'intérieur des réunions. Il semble donc que la nature du rôle et l'expérience de la personne jouant ce rôle aient contribué à ce genre d'interaction.

SOMMAIRE

Il n'existe pas beaucoup de données empiriques qui permettent d'évaluer l'efficacité des méthodes de développement logiciels. On sait cependant que les réunions occupent un place prépondérante dans le cheminement d'une équipe vers son but final, le développement d'un logiciel. Il est donc important de mieux comprendre les phénomènes de communication présents lors du développement de logiciels par une équipe de professionnels. Ce sommaire des principales recherches sur la solution de problèmes en groupes a permis de déterminer trois domaines distincts:

1. La formation des équipes de travail performantes;
2. Les techniques de solution de problèmes en groupe et la dissémination de la connaissance par les questions et les réponses des développeurs lors des réunions; et
3. L'organisation de l'équipe de développement, son leadership e le rôle joué par les membres de l'équipe.

Nous avons pu voir que la solution de problèmes en groupe est un processus complexe en lui-même et que son efficacité dépendra en grande partie de la connaissance de ses techniques par les membres du groupe de décision. Il est facile de comparer un groupe type de solution de problèmes avec un groupe d'ingénieurs logiciels tentant d'élucider un problème complexe lord d'une réunion d'analyse ou de design. Dans la même

veine, une réunion d'inspection technique peut être considérée comme une réunion de prise de décisions durant laquelle un groupe d'individus doit se mettre d'accord sur l'implantation d'un système avant de commencer la programmation.

La communication est la pierre angulaire du travail en équipe. Un manque de communication ne peut que rendre le travail en groupe inefficace et peut-être inutile. Les utilisateurs du processus de génie logiciel ne peuvent que bénéficier d'une meilleure connaissance des méthodes de communication utilisées dans leur travail de tous les jours.

Les données recueillies par cette recherche pourront être utilisées pour augmenter nos connaissances existantes de l'aspect organisationnel du développement logiciel et de l'efficacité des réunions présentes tout au long du processus. De plus, Olson et Herbsleb ont mentionné que le rôle joué par les membres de l'équipe lors des réunions est un aspect important mais qu'il n'a pu encore être mesuré. D'un autre côté, Thomsett décrit un modèle (Belbin) qui permet d'établir les rôles nécessaires à la bonne marche d'un groupe. Il est bien vrai que l'efficacité d'une équipe est basée sur les membres qui en font partie, mais il s'agit de déterminer si les rôles (postes) qu'ils jouent dans l'organisation ont un impact sur le contenu et la fréquence des conversations. Une étude similaire à celle de Herbsleb, tout en y ajoutant un élément de rôle à chacun des individus pourraient peut-être amené des résultats intéressants.

Le projet de développement de logiciels qui est sous observation est un projet ayant des contraintes de budget, d'échéanciers et de qualités. De plus, l'équipe consistera d'analyste-programmeurs ayant une formation adéquate et un niveau d'expérience suffisant. Les données recueillies durant la période d'observation pourront donc être utilisées avec d'autres groupes de développement de logiciels puisqu'il s'agit d'un projet ayant toutes les caractéristiques retrouvées dans d'autres situations de la vie réelle.

Bref, il existe un grand nombre d'outils permettant de mesurer l'efficacité d'un groupe et il est donc nécessaire de tenter de comprendre les caractéristiques distinctes des groupes de développement de logiciels. Une recherche comme celle présentée ici pourrait

accomplir une partie de ce travail.

PARTIE 2 - PROTOCOLE DE RECHERCHE

CONCEPTUALISATION DU PROBLÈME DE RECHERCHE

Objectif général de la recherche

Les processus de Génie Logiciel actuels permettent aux ingénieurs d'exécuter les tâches nécessaires à la conception d'un logiciel dans une suite d'étapes logiques facilement identifiables et mesurables. La communication permet à une équipe d'ingénieurs logiciels de partager ses connaissances et ses résultats afin de réussir le projet de développement. Un élément important du processus de conception de logiciels est donc le partage de l'information à l'intérieur de l'équipe. Pour ce faire, les processus de développement utilisés présentement se basent sur plusieurs types de réunions afin de bâtir un cadre efficace pour l'échange de l'information. Cette recherche a donc pour but d'identifier le contenu de cette information en fonction de différentes caractéristiques des réunions.

Cible de la recherche

Cette recherche tentera de décrire et d'expliquer les phénomènes de communication qui agissent d'une façon ou d'une autre sur le processus de Génie Logiciel. Elle vise à mieux outiller les ingénieurs logiciels afin d'améliorer leur rendement et aussi celui de l'organisation pour laquelle ils travaillent.

Ce projet permettra de définir les caractéristiques de la communication à l'intérieur du processus logiciel et d'identifier son contenu afin d'amener des alternatives aux processus logiciels actuels.

Importance du sujet de la recherche

Chaque individu approche la tâche "d'écrire un programme" avec son propre bagage de connaissances et d'expérience. Il faut donc supposer que la communication de

l'information à l'intérieur de l'équipe est l'élément qui harmonise ces connaissances et permet l'utilisation de l'expertise de chacun des membres par l'équipe.

Le processus lui-même prévoit l'utilisation de plusieurs périodes allouées à l'échange d'information et à la discussion d'idées afin de faire avancer le projet. L'identification et l'évaluation de ces instruments dans le processus de génie logiciel est donc un travail important et nécessaire.

Il n'existe pas beaucoup de données empiriques qui permettent d'établir avec certitude le contenu de l'information communiquée dans un projet de développement logiciel. Cette recherche aidera à mieux identifier le contenu de l'information présent lors du développement de logiciels par une équipe de professionnels.

Le génie logiciel demande de chaque individu des aptitudes à pouvoir imaginer et créer des entités abstraites et surtout à être capable de relier ces entités ensemble grâce à des liens qui eux aussi sont abstraits. Ce sont ces caractéristiques qui rendent le travail en équipe très difficile car il n'existe pas deux personnes qui imaginent la même solution à un problème très précis. Le génie logiciel vient en aide à ces gens en leur donnant des processus et des outils de mesures qui permettent de mieux comprendre leur rôle dans le développement d'un logiciel. Ces processus de Génie logiciel sont des outils de travail qui ont été créés dans le but de donner un cadre relativement rigide de travail aux ingénieurs et en leur permettant de visualiser les étapes de développement d'un logiciel. Une organisation type adaptera un modèle reconnu pour ses propres besoins et ceux des membres de l'organisation.

Formulation du problème de recherche

L'utilisation d'équipes pour concevoir et implanter un logiciel n'a rien de nouveau. Le processus logiciel permet de structurer le travail d'une équipe de développement afin d'obtenir du travail efficace et des résultats qui sont répétés à chacun des projets. Ce processus est donc le point de départ de toutes les activités retrouvées dans un projet de développement de logiciels. La communication à l'intérieur de l'équipe de

développement est peut-être dépendante du processus employé. L'information qui y est communiquée est souvent faite en fonction de l'étape présent du processus. Il existe cependant plusieurs façons de disséminer l'information à l'intérieur d'un équipe. Les discussions formelles se font dans le cadre de plusieurs types de rencontres sanctionnées par le processus de travail. Deux grandes classes de réunions sont couramment utilisées, soient les réunions de travail et de révisions. Il s'agit de déterminer l'influence de ces différentes classes sur le contenu de l'information échangé.

Stade de développement de la recherche

Une recherche exploratoire a été commencée il y a quelques temps dans le but de valider les données recueillies ainsi que nos méthodes d'observations. Cette recherche exploratoire consiste à reproduire une étude d'Herbsleb [29] et de déterminer si les résultats obtenus sont comparables aux siens.

Les objectifs de ce travail préparatoire sont nombreux. Il permet d'augmenter notre expérience dans la recherche expérimentale en se servant de résultats existants comme base afin de déterminer les aspects importants du travail ainsi que ceux qui nécessiteront des améliorations dans la suite de la recherche. Par la même occasion, ce travail permet de valider les résultats d'Herbsleb en vérifiant que ceux-ci sont applicables à d'autres équipes de travail que celles observées dans sa recherche.

Utilisation des résultats de la recherche

La communication est la pierre angulaire du travail en équipe. Un manque de communication ne peut que rendre le travail en équipe inefficace et peut-être inutile. Les utilisateurs du processus de génie logiciel ne peuvent que bénéficier d'une meilleure connaissance des méthodes de communication utilisées dans leur travail de tous les jours.

Les données recueillies par cette recherche pourront être utilisées pour augmenter nos connaissances existantes de l'aspect organisationnel du développement logiciel.

Modèle théorique retenu

La communauté informatique s'accorde pour dire que l'amélioration du processus de développement de logiciels amène une amélioration de la qualité du produit. Depuis 1985, plusieurs différents types de processus ont été proposés dans le but de répondre aux besoins particuliers des organisations de développement de logiciels, et ce, dans le but manifeste d'améliorer la qualité du produit final. Il s'agit maintenant de se poser la question si les processus envisagent toutes les situations et les variables qui agissent sur la qualité du logiciel ou s'il en existe d'autres qui sont aussi importantes.

La première partie de ce document a permis de mettre en lumière qu'il existe plusieurs facettes du développement de logiciels que les processus actuels ne prennent pas en considération. L'importance de la communication comme instrument de dissémination et d'harmonisation de l'information à l'intérieur de l'équipe de travail est l'une d'elle. Il s'agit maintenant de mieux comprendre les principales caractéristiques de cette communication en identifiant son contenu.

Le rationnel d'une communication permet d'en discerner le contenu selon certaines catégories afin d'établir plus précisément les points importants discutés par les participants. Il existe plusieurs types de rationnels tels que celui d'Olson et al. [30] qui ont établi un nombre de catégories permettant de modéliser la nature de l'information échangée lors de réunions de design. Cette technique permet d'éliminer les données superflues et de trouver des patrons dans le contenu de l'information lors d'une réunion. Le rationnel est donc le "squelette" de la communication.

Le type de communication utilisé durant le processus de développement de logiciels peut être des réunions formelles ayant trait au design ou à la révision de technicalités du projet ainsi que toutes documentations qui découlent de ces réunions. Les échanges informels seront aussi considérés comme un type de communication.

Hypothèse de recherche

Afin de bien observer une situation de la vie réelle et d'obtenir des données qui pourront être utilisées dans le futur, il s'agit de bien définir et préciser l'hypothèse de recherche qui sera utilisée tout au long du présent projet. Dans ce projet de recherche, l'hypothèse nulle qui énonce formellement les relations attendues entre la nature de la communication et son rationnel est la suivante:

La rationnel d'une communication est constant en fonction de la durée, du sujet et du type de communication présents dans le processus de développement de logiciels.

STRATÉGIE DE RECHERCHE

Cette section tentera de décrire la stratégie de recherche adoptée pour vérifier l'hypothèse décrite auparavant et de vérifier la validité de cette stratégie.

Approche générale

Le travail envisagé dans ce document sera accompli grâce à une recherche expérimentale qui visera, en utilisant de façon systématique les connaissances existantes et en modifiant la variable indépendante (la nature de la communication), à évaluer ses effets sur la variable dépendante (son rationnel) et ainsi à mieux comprendre la communication présente dans le processus de développement de logiciels.

Il ne s'agit pas de modifier intentionnellement ou de contrôler parfaitement la variable indépendante, mais plutôt d'observer les agissements des individus du groupe échantillon lors de l'utilisation des différents types de communication. Pour cette raison, l'utilisation de l'expérimentation invoquée dans laquelle la variable indépendante n'est pas manipulée mais plutôt qui varie de façon naturelle ou accidentelle et une méta-analyse, c'est-à-dire l'application de méthodes statistiques afin d'intégrer et

synthétiser les résultats de plusieurs observations du groupe expérimental sont envisagées.

Validité de la stratégie de recherche

La validité interne sera garantie par l'efficacité à parvenir à éliminer toutes les explications plausibles autres que l'influence de la variable indépendante sur la variable dépendante. Il y aura biaisage des données si la recherche effectuée ne permet pas d'éliminer l'influence de toutes les variables autres que celle qui est indépendante.

La variable indépendante pourra varier selon trois caractéristiques bien précises: la durée d'une communication, le type de communication et le sujet de la communication. Chacune des ces variations sera observée afin d'obtenir des résultats qui sont statistiquement justifiés.

Dans le cas de cette recherche, le groupe à l'étude est constitué de quatre ingénieurs logiciels qui travaillent ensemble depuis près d'un an. Durant cette période, ils sont devenus confortables avec le processus de développement de logiciels utilisés dans leur organisation. Leur projet comporte toutes les caractéristiques d'un projet de développement de logiciel type, c'est-à-dire la difficulté constante de pouvoir coordonner les besoins de l'échéancier, le budget et la qualité du produit. De plus, les quatre individus sont maintenant habitués au travail d'équipe requis par ce projet et à l'interaction demandée. En somme, un changement de la variable dépendante ne devrait s'effectuer que par un stimuli de la variable indépendante.

Afin de s'assurer que cette recherche fournira des résultats généralisables à d'autres populations, il faut se demander jusqu'à quel point nos résultats seront influencés par le contexte particulier dans lequel travail notre groupe expérimental.

Le projet de développement de logiciels qui sera sous observation sera un projet ayant des contraintes de budget et d'échéanciers. De plus, l'équipe consistera de concepteurs expérimentés (des professeurs universitaires) et d'ingénieurs logiciels ayant une

formation adéquate et un niveau d'expérience suffisant à la bonne marche du projet.

Les résultats obtenus à partir des données recueillies durant la période d'observation pourront être utilisées avec d'autres équipes de développement de logiciels puisqu'il s'agit d'un projet ayant toutes les caractéristiques retrouvées dans d'autres situations.

Population cible

Le domaine de l'informatique est constitué d'individus ayant de grandes connaissances dans la programmation et les différents paradigmes couramment utilisés.

Échantillon

L'échantillon utilisé dans cette recherche est une équipe de projet qui sera composée d'ingénieurs logiciels ayant une grande connaissance des technologies et techniques nécessaires pour la réalisation du produit. Cependant, leur connaissance du domaine d'application du produit est peut-être limitée à leur interaction et les explications des représentants du client. Tout comme n'importe quel autre projet similaire, c'est la communication qui permettra d'uniformiser les connaissances et l'information nécessaire à la bonne marche du projet.

Les ingénieurs logiciels de l'équipe sont des individus ayant une formation en informatique. De plus, la première partie du projet a été utilisée pour former les nouveaux arrivants aux procédures utilisées par l'équipe. Il existait donc une disparité quant à l'expérience des individus, ce qui constituait une situation similaire retrouvée dans l'industrie de l'informatique.

Variable dépendante

Le rationnel d'une communication sera identifié grâce à la modélisation des données recueillies lors de l'occurrence de la variable indépendante (réunions de design, inspection technique, échanges informels). Dans le cas du projet qui sera observé, les données suivantes seront disponibles: les enregistrements et les transcriptions de

réunions de design et d'inspection technique et la documentation utilisée ou produite par les réunions observées.

Variable indépendante

Le type de communication ne pourra être contrôlée en soi, mais il sera important de pouvoir identifier les relations qui existent entre ses caractéristiques et la variable dépendante afin d'en discerner les éléments importants et de pouvoir former une théorie qui explique correctement l'apport de la communication dans le processus de Génie logiciel.

Méthode de collecte des données

Il existe trois grandes sources de données: l'utilisation de documents, l'observation par le chercheur et l'information fournie par les sujets. Dans le projet qui nous intéresse, le processus de Génie logiciel utilisé par le groupe échantillon nous permet d'obtenir des données intéressantes sur les sujets suivants:

- L'enregistrement de diverses réunions de design et d'inspection technique ainsi que leur transcription permettront d'identifier les différents sujets discutés ainsi que de faciliter l'étude de l'interaction et la dynamique de groupe présentes lors de ces réunions grâce à la technique des rationnels.
- Un système de jetons permettra de comptabiliser l'emploi du temps de chacun des membres de l'équipe afin de déterminer, entre autre, l'effort requis à la communication et les résultats d'une meilleur qualité dans la communication de l'équipe.

Dans le cas de l'observation de l'équipe lors des réunions, l'observation systématique sera utilisée, c'est-à-dire que le processus de travail normal de l'équipe ne sera pas modifié ou biaisé par notre observation. Il y aura donc une grande distance entre le chercheur et le phénomène à observer.

Qualité des instruments de mesure

La fiabilité d'un instrument de mesure est sa capacité à reproduire un résultat de façon consistante dans le temps et dans l'espace. Il faut remarquer qu'un instrument qui est faussé de façon systématique peut être parfaitement fiable mais manquer de validité.

Il sera très important de s'assurer que les mesures du contenu de l'information soient fiables parce que la non-fiabilité de celles-ci réduiraient la possibilité d'observer un effet significatif d'une variation du rationnel de la communication et, par conséquent, la puissance de la recherche.

Afin de mesurer la fiabilité de nos instruments de mesure, nous utiliserons plusieurs approches dont la comparaison des résultats obtenus par l'utilisation du même instrument à différents moments (dans le cas de la transcription de réunions ayant les mêmes caractéristiques mais situés à différents moments lors du projet), pour évaluer sa stabilité. L'évaluation des données selon les différents critères que nous amènerons sera faite par plusieurs observateurs à un même moment afin d'assurer une fiabilité inter-juges.

Validité de la mesure

La validité se définit comme la capacité d'un instrument à mesurer le phénomène étudié, c'est-à-dire l'adéquation qui existe entre les variables retenues et le concept théorique à mesurer. Il existe une forme de validation appelée la validation de trait qui consiste à corréler les résultats obtenus par l'observation à valider avec les résultats obtenus par un autre test mesurant sensiblement le même phénomène. La corrélation des résultats donnera une indication de validité du modèle théorique qui sera parfaitement symétrique. Cette forme de validation sera obtenue dans les premières phases de la recherche grâce à la comparaison des résultats avec ceux obtenus par les recherches d'Herbsleb.

Analyse des données

Les données qualitatives proviendront de l'observation de l'équipe de travail lors des réunions de design et d'inspection technique faisant partie intégrale du processus de développement qu'ils utilisent. L'observation initiale sera faite à l'aide d'une caméra vidéo que les participants actionneront eux-mêmes au début et à la fin des réunions. Le texte des réunions sélectionnées sera transcrit dans un document qui devra reproduire le mieux possible toutes les interactions entre les participants de chaque réunion. Ces transcriptions seront disponibles pour effectuer plusieurs recherches et analyses dans le futur.

L'étape suivante permettra d'analyser les données à l'aide du modèle théorique (un rationnel) afin de modéliser les différentes configurations empiriques observées. C'est durant cette période que les observations seront décortiquées de façon à trouver une ou des relations entre nos variables dépendantes et indépendantes. Une analyse transversale sera utilisée par la suite pour s'assurer qu'il y a réplique des résultats parmi plusieurs cas ou situations.

SOMMAIRE

Cette recherche permettra d'identifier le contenu de l'information qui est véhiculée par différents types de communication présents dans le processus de développement de logiciels. De plus, les résultats de ce travail devraient donner un rapport précis de l'effort requis pour la communication à l'intérieur d'un projet de Génie Logiciel et ainsi, donner une aide précieuse lors de la planification et la gestion d'autres projets similaires.

Tous ces résultats devraient nous donner une image précise de l'importance de la communication dans les processus logiciels et ainsi permettre l'élaboration de règles simples que les organisations de développement de logiciels pourront adopter.

Enfin, il est important de mentionner qu'au meilleur de notre connaissance, aucun travail publié jusqu'à maintenant ne traite de l'analyse des types de communication

présents dans un processus logiciel dans le but de comparer leur contenu en fonction de leur durée, type et sujet. Ce travail de recherche constitue donc en soi un projet novateur. Cette contribution scientifique permettra aux ingénieurs logiciels de mieux comprendre l'impact de la communication dans le développement de logiciels. Donc, avec les résultats initiaux très prometteurs obtenus, il serait souhaitable de poursuivre ce projet de recherche immédiatement au niveau doctoral afin de préserver l'originalité et la globalité du projet.

RÉSULTATS INITIAUX OBTENUS

Les travaux initiaux nous ont permis d'évaluer notre méthode d'analyse des données. Quelques réunions d'inspection technique et d'analyse ont été observées par l'entremise de d'enregistrement vidéo et de transcription des conversations entre les différents intervenants de ces réunions. Une analyse similaire à celle effectuée dans les travaux d'Herbsleb a permis de comparer nos méthodes de travail et de les améliorer en fonction des nombreux conseils qu'Herbsleb lui-même nous a fournis. Le diagramme de la figure 3 représente les résultats initiaux obtenus à la suite de nos analyses de la réunion de design du 28 mars 96.

Les différents cercles sont une représentation proportionnelle du temps consacré à chacune des catégories. Par exemple, la catégorie des critères, qui désigne les raisons, arguments ou opinions qui évaluent une solution alternative ou une proposition, a été la plus présente lors de cette réunion, suivie des alternatives, qui désignent les solutions ou propositions au sujet des caractéristiques pouvant être offertes à l'utilisateur ou des façons d'implanter des caractéristiques préalablement choisies. Les flèches désignent les transitions entre les différentes catégories. Elles sont une représentation proportionnelle des changements de catégories qui ont eu lieu au cours de la réunion. Les plus importantes sont celles critères-critères et alternatives-alternatives. Il faut prendre note que le graphique ne montre que les transitions dont le montant était plus élevé que 1% du total des transitions de cette réunion.

Error! Not a valid link. **Figure 3 Résultats de l'analyse de type Herbsleb d'une réunion de design**

En gros, cela signifie que le contenu de la réunion a tourné autour de la proposition de plusieurs alternatives qui ont été débattues par les membres de l'équipe de développement. Une réunion de design permet donc à un groupe de concepteurs logiciels d'émettre des hypothèses et de les débattre entre eux. Cependant, le graphique démontre aussi qu'une bonne partie du temps est consacré à d'autres types d'activités tels que la gestion de la réunion, la gestion du projet, etc. Cela ne répond pas à toutes les

questions puisqu'il n'est pas possible, à partir de cette analyse seulement, de savoir le nombre d'alternatives et de critères qui étaient présents lors de cette réunion.

L'analyse du rationnel de la réunion nous permet de représenter ces activités (Voir la figure 4). À l'aide de ces deux analyses, il nous est maintenant possible de savoir le ratio entre le temps passé dans les catégories d'alternatives ou de critères et le nombre de ceux-ci générés durant la réunion. Une analyse plus approfondie nous permettrait de déterminer l'efficacité de ce type de réunions ainsi que leur influence sur le projet lui-même.

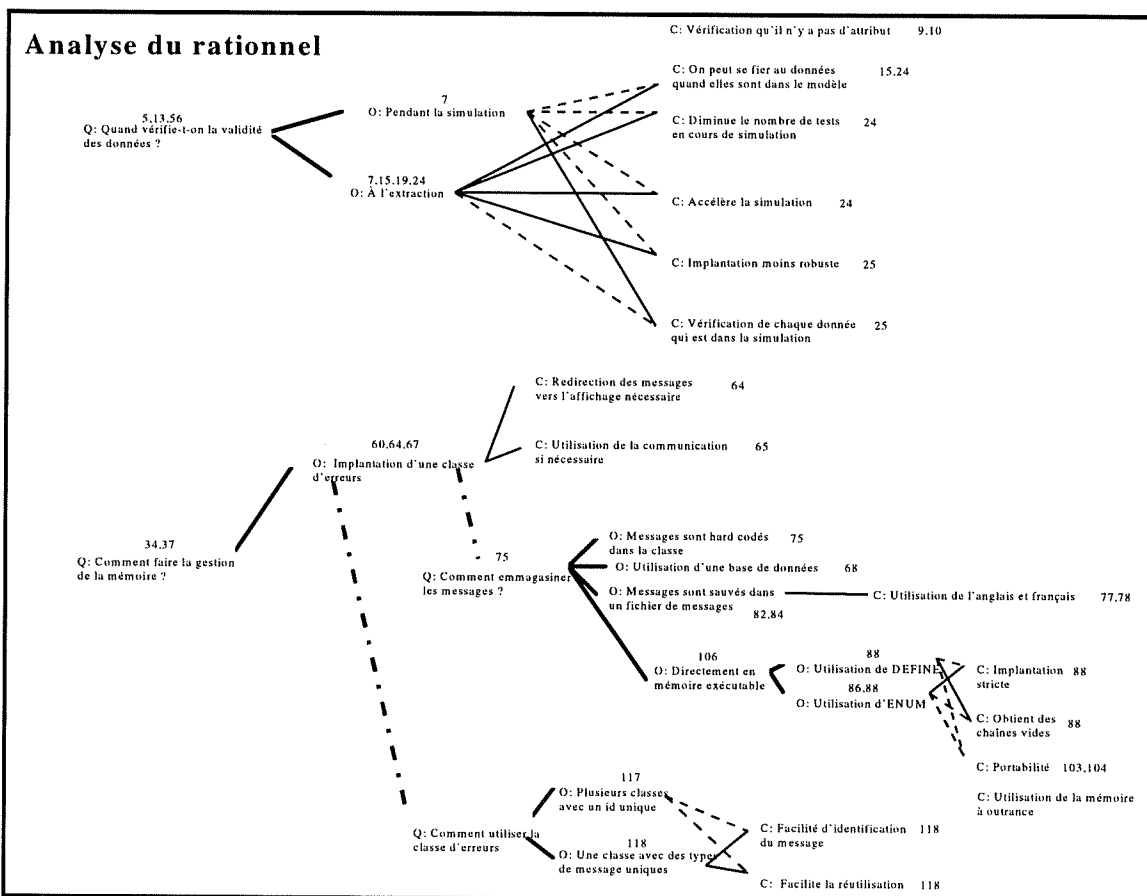


Figure 4 Analyse du rationnel de la réunion de design

Encore une fois, il s'agit de résultats initiaux qui avaient pour but de s'assurer de la qualité de notre démarche expérimentale. Cependant, elle nous a déjà permis d'observer plusieurs caractéristiques intéressantes de ce type de réunions. La suite de nos

travaux nous permettra d'élargir nos observations à d'autres réunions de design et d'inspections techniques.

TRAVAUX FUTURS

Les travaux de recherches sont déjà commencés afin d'obtenir un nombre suffisant de données (les transcriptions de réunions) pour commencer l'analyse de leur rationnel. De plus, nos connaissances actuelles du problème nous permettent de mieux définir nos besoins en données et de mieux diriger nos observations dans les prochains mois. Cela coïncide avec la dernière phase du projet de développement échantillon.

Les données obtenues auparavant nous indiquent que la méthode d'observation est valide et nous permet d'obtenir des résultats fiables, grâce à leur comparaison avec ceux des travaux préalables d'Herbsleb. Voici les principaux aspects qui seront étudiés dans les prochaines phases de cette recherche:

- une analyse de réunions de design et d'inspection technique traitant d'une même fonctionnalité (sujet) afin de comparer leur contenu;
- une analyse de plusieurs réunions de design afin de déterminer, grâce au rationnel, s'il existe une similitude quant à leur contenu;
- une analyse similaire pour les réunions d'inspection technique; et
- une analyse du rationnel sera effectué sur des réunions de même type mais ayant des durées différentes afin de déterminer si la longueur d'une réunion a un effet sur le contenu.

COLLABORATION

La bonne marche de ce projet demande des connaissances dans plusieurs disciplines autres que celle du Génie Logiciel. Afin de s'assurer d'obtenir des résultats de qualité et

significatifs, plusieurs chercheurs ont offert leur collaboration.

Dans la phase initiale du projet, qui consiste à s'assurer de la fiabilité de nos données ainsi que de nos méthodes d'analyse, nous avons reçu l'aide de Jim Herbsleb du Software Engineering Institute (SEI). Sa collaboration nous a permis de sauver un temps précieux lors du démarrage du projet. Il nous a fourni beaucoup d'information au sujet de son expérience préalable.

Afin d'améliorer nos connaissances en psychologie et dans les protocoles des réunions de groupe, Mme Aude Dufresne du Département de communication de l'Université de Montréal a fourni son aide au projet et est prête à continuer dans le futur.

Finalement, un projet d'échange France-Québec a été initié afin de faire converger nos efforts avec ceux du groupe de recherche de Mme Françoise Détienne de l'Institut National de Recherche en Informatique et en Automatique (INRIA). Cette collaboration consistera à étudier les aspects cognitifs et les protocoles de communication présents lors des réunions d'inspections. L'expérience de l'INRIA nous sera très utile pour l'avancement de notre projet.

CONCLUSION

La première partie de ce document a permis de démontrer que les aspects de l'échange de l'information dans le processus de Génie logiciel est un élément important que l'on doit tenir compte afin d'améliorer la façon de faire d'une organisation dans le but ultime d'augmenter la qualité des logiciels produits par cette organisation. Cependant, il existe plusieurs caractéristiques qui définissent cet échange d'information: la formation des équipes, les techniques de solutions de problèmes en groupe et le rôle des membres dans le groupe et aussi dans les conversations. La communication est le mécanisme qui permet d'harmoniser tous ces éléments dans un tout utilisable par les personnes qui doivent gérer le groupe et les projets de développement de logiciels.

Il n'est cependant pas très clair, aujourd'hui, jusqu'à quel point la communication est bien utilisée dans ces groupes de développement et si l'effort mis sur son utilisation rapporte les dividendes escomptés. C'est ce que ce travail cherche à découvrir par l'observation d'un groupe d'ingénieurs logiciels typique qui travaillent sur un projet de développement de logiciels semblables à plusieurs autres pour un client ayant des priorités normales. Ce sont ces caractéristiques du groupe échantillon qui rendent ce travail d'observation intéressant: il s'agit d'une situation naturelle, tout à fait interchangeable avec plusieurs autres présentes actuellement dans la communauté informatique.

Le protocole de recherche qui est décrit dans la deuxième partie de ce document est encore très large, il n'explique pas en détails chacune des études que nous comptons faire à partir des données recueillies. La recherche initiale consiste à reproduire un travail accompli par un autre groupe de recherche dans le but de s'assurer que nos méthodes d'observations et d'analyses sont valides et fiables et, par la même occasion, peut-être compléter le travail d'Herbsleb. La section précédente a permis de démontrer que les recherches sont déjà en cours et que des résultats préliminaires nous permettent de croire du bien fondé de notre démarche.

L'utilisation d'un processus de génie logiciel est la clé qui permettra à une organisation de développement de logiciels de s'améliorer et ainsi produire des logiciels de qualité. Ces processus contiennent une série d'étapes à suivre et un nombre d'objectifs à atteindre afin d'obtenir un niveau acceptable de qualité dans le produit. D'un autre côté, ces processus ne donnent pas les outils nécessaires afin de franchir ces étapes.

L'étude de l'échange d'information dans le développement de logiciels répondra à plusieurs questions des ingénieurs logiciels et complétera les processus de développement actuels. Plus précisément, l'observation d'équipes d'ingénieurs logiciels nous fournira des données extrêmement utiles sur la façon dont les logiciels sont conçus actuellement. Une analyse de ces données nous donnera une image précise des manques à gagner dans

les processus logiciels actuels et pourra aider à trouver des solutions à ces problèmes.

Les données disponibles à partir des enregistrements vidéos sont de trois types: (1) des réunions de design où les ingénieurs amènent de nouvelles idées, les critiquent et décident ce qui sera, à la fin, quelques unes des caractéristiques du logiciel, (2) des réunions d'inspection technique où les membres du groupe évaluent et valident un document d'analyse écrit par l'un des coéquipiers et (3) des réunions de gestion qui permettent à l'équipe de s'informer de tout qui se rapporte à la gestion du projet. Une analyse intéressante sera la comparaison de deux réunions portant sur le même sujet mais à deux temps différents: une réunion de design et une réunion d'inspection technique. Cet exercice pourrait faire la lumière sur l'information qui est conservée lors d'une réunion de design jusque dans le document final validé par la réunion d'inspection technique.

La dernière partie de ce travail, mais non la moindre, serait de tenter modéliser cet échange d'information afin de déterminer les bonnes et mauvaises habitudes que l'on retrouve dans le travail d'une équipe de développement.

RÉFÉRENCES

1. J.E. Tomayo, " The Airplane as Computer Peripheral." , *American Heritage of Inventions & Technology*, hiver 1991.
2. J. Kanter, S. Schiffman et J.F. Horn, " Let the customer do it; from grocery robots to photo kiosks, comuterized self-service is in." , *Computerworld*, Août 1990, pp 24-35.
3. R. Woolnough, " TI drives at Euro autos; makes 'PACT' to win microcontroller business." , *Electronic Engineering Times*, Août 1991.
4. Comptroller General, " Contracting for Computer Software Development." , FGMSD-80-4, Government Accounting Office, septembre 1979.
5. S.R. Schach, *Software Engineering*, Aksen Associates, 1990.

6. The PMI Standards Committee, *Project Management Body of Knowledge*, Project Management Institute, 1987.
7. F.P. Brooks, "No silver bullet", *Computer*, avril 1987, pp 10-19.
8. R.L. Van Scoy, "Software Development Risk: Opportunity, Not Problem". Software Engineering Institute, Carnegie Mellon University, 1992.
9. M.A. Berger, "The Technical Approach to Teamwork", *Training and Development Journal*, mars 1985, pp 53-55.
10. J. Dewey, *The influence of Darwin on Philosophy, and other essays on contemporary thoughts*, Bloomington, Indiana University Press, 1910.
11. R.F. Bales, *Interaction Process Analysis: A method for the Study of Small Groups*, New York, Holt.Rinehart and Winston, 1950.
12. J.M. Scheidel, *Speech communication and Human interaction*, Glenview, Il, Scott. Foresman, 1976.
13. A.B. Fisher, *Small Group Decision Making: Communication and Group Process*, Chapitre 7, New York, McGraw Hill, 1974.
14. M. Doyle, D. Strauss, *How to Make Meetings Work*, New York: Jove Books, 1976, Chapitre 16.
15. W.R. Reitman, *Cognition and thought*, New York:Wiley, 1965.
16. G.M. Olson, J.S. Olson, M.R. Carter, M. Storosten, "Small Group Design Meetings: An Analysis of Collaboration", *Human-Computer Interaction*, Vol. 7, 1992, pp 347-374.
17. R. Thomsett, "Effective Project Teams.", *American Programmer*, juillet/août 1990, pp 25-35.
18. F.E. Emery, M. Emmery, *Participative Design-work and Community*, ACT, Australian Centre for Continuing Education, Australian National University, Canberra, 1975.
19. J.R. Hackman, G.R. Oldham, *Work Redesign*, Reading, MA:Addison-Wesley, 1980.
20. I.B. Myers, *Gifts Differing*, Palo Alto, CA:Consulting Psychologists Press, 1980.

21. M.L. Lyons, "The DP Psyche", *Datamation*, août 1985, p. 103.
22. R.P. Bostrom, K.M. Kaiser, "Personality Differences within Project Teams", *Proceedings of the 18th Annual Computer Personnel Research Conference*, ACM No. 443810, 1981.
23. C.M. Bush, L.L. Schkade, "In Search of the Perfect Programmer", *Datamation*, mars 1985, p.128.
24. R.M. Belbin, *Management Teams*, London:Henimann, 1981.
25. F.P. Brooks, *The mythical man-month: Essays on software engineering*, Reading, MA:Addison-Wesley, 1975.
26. J. Herbsleb, "OOD and Software Design Teams", *Human-Computer Interaction*,
27. G. Booch, *Object oriented design with applications*, Cambridge, MA:Harvard University Press, 1991.
28. J.E. McGrath, *Groups: Interaction and performance*, Englewood Cliffs, NJ:Prentice-Hall, 1984.
29. J. Herbsleb, "OOD and Software Design Teams", *Human-Computer Interaction*,
30. G.M. Olson, J.S. Olson, M.R. Carter, M. Storosten, " Small Group Design Meetings: An Analysis of Collaboration", *Human-Computer Interaction*, Vol. 7, 1992, pp 347-374.

ÉCOLE POLYTECHNIQUE DE MONTRÉAL



3 9334 00171573 7

École Polytechnique de Montréal
C.P. 6079, Succ. Centre-ville
Montréal (Québec)
H3C 3A7

