



HAL
open science

L'externalisation du développement d'applications

Marie-Hélène Delmond

► **To cite this version:**

Marie-Hélène Delmond. L'externalisation du développement d'applications. Gestion et management. HEC PARIS, 1994. Français. NNT : 1994EHEC0026 . pastel-00994919

HAL Id: pastel-00994919

<https://pastel.hal.science/pastel-00994919>

Submitted on 22 May 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ECOLE DES HAUTES ETUDES COMMERCIALES
JOUY-EN-JOSAS

THESE

POUR L'OBTENTION DU TITRE
DE DOCTEUR ES SCIENCES DE GESTION

Nouveau doctorat (arrêté du 30 mars 1992)

SUJET

L'EXTERNALISATION DU DEVELOPPEMENT D'APPLICATIONS

Candidate

MARIE-HELENE DELMOND

JURY

Président

ROBERT REIX

Professeur, Directeur honoraire, Responsable du CAAE à l'Université de Montpellier II

Suffragants

JEAN-LOUP ARDOIN
Professeur au Groupe HEC

GENEVIEVE CAUSSE, rapporteur
Professeur à l'Université de Paris XII, Centre Saint-Simon

YVES DUPUY

Professeur à l'IAE de l'Université de Montpellier II, rapporteur

SERGE LACRAMPE

Professeur au Groupe HEC, directeur de thèse

JEAN-MICHEL MATHIEU
Associé, Cabinet Arthur Andersen

Année de soutenance : 1994

Le Groupe HEC n'entend donner aucune approbation ni improbation aux opinions émises dans cette thèse : ces opinions doivent être considérées comme propres à l'auteur.

Mon voisin (ancien HEC à la retraite), l'oeil malicieux :

- "Alors, cette thèse, ça avance ?

- ...

- Et votre mari, qu'en dit-il ?

- Il m'encourage !

- Alors, ça va. Vous savez, j'ai l'habitude, quand un collaborateur me disait "je veux poursuivre des études en parallèle", je lui disais, "oui, mais, est-ce que votre femme est d'accord ?" En fait, c'est le couple qui se lance dans le projet !"

A Pierre, pour son indéfectible soutien.

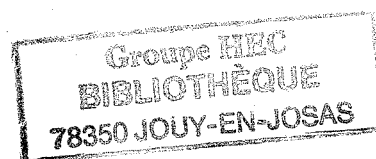
Remerciements

Mes remerciements s'adressent en premier lieu à mon Directeur de Recherche, Monsieur Serge Lacrampe, pour ses encouragements, son dynamisme et ses conseils judicieux, et pour la façon dont il a su me guider tout en respectant mon travail.

Je veux aussi remercier en tant qu'institution le Groupe HEC et la Fondation HEC de m'avoir fait confiance et de m'avoir assuré des conditions favorables à l'aboutissement de cette thèse. Mes remerciements s'adressent tout particulièrement à Patrick Le Maire, coordinateur du département systèmes d'information et d'aide à la décision où j'enseigne depuis trois ans ; qu'il trouve ici l'expression de ma très sincère gratitude.

Je remercie également tous ceux qui m'ont soutenue et encouragée dans l'élaboration de cette thèse, en particulier mes collègues des départements systèmes d'information et contrôle de gestion. Cette double filiation a été pour moi une source d'enrichissements et a profondément marqué l'orientation de mes travaux. Des remerciements tous particuliers s'adressent à Michel Tenenhaus et Jacques Obadia, pour les conseils et l'aide qu'ils m'ont prodigués et à Michel Lebas, qui m'a initiée aux subtilités de la théorie de l'agence. Enfin, ma reconnaissance s'adresse spécialement à Jean-Loup Ardoin dont la rigueur, le regard tout à la fois critique et bienveillant et l'attention constante m'ont permis de faire évoluer ce travail dans un sens plus proche de ce que je souhaitais.

J'adresse également mes remerciements à tous ceux qui, à des titres divers, ont témoigné de l'intérêt pour cette recherche et facilité sa réalisation : Michel Ghertman, professeur au groupe HEC, Michel Léger et Jean-Michel Mathieu au cabinet Arthur Andersen, ainsi ceux et celles qui m'ont accueillie et accordé du temps dans le cadre de mes enquêtes terrain.



Réaliser une thèse tout en menant des activités professionnelles d'enseignement est un exercice périlleux sur le plan de l'organisation personnelle et familiale : sans le support constant de mon entourage, l'entreprise eût été impossible. Avoir l'assurance de savoir mes enfants entourés de l'affection de leurs grands parents et bénéficiant du sain climat breton m'a certes considérablement aidé à transformer certaines périodes de vacances scolaires en moments privilégiés de recherche et d'écriture !

Enfin, je remercie vivement Huguette Le Moal et Caroline De Franco pour leur aide dans la prise en charge de la réalisation matérielle de ce document. Que leur disponibilité, leur gentillesse et leur professionnalisme soient ici remerciés.

L'externalisation du développement d'applications

Résumé

Quelles sont les activités que l'entreprise doit prendre en charge elle-même et celles pour lesquelles il est plus judicieux de faire appel à l'extérieur ? A l'heure même où se développent les organisations en réseau et la pratique du partenariat, cette question apparaît comme l'un des thèmes majeurs du management. Ce débat est particulièrement sensible dans le domaine informatique, où se sont multipliés ces dernières années les exemples d'entreprises ayant décidé de cesser d'assurer elles-mêmes la gestion directe de leurs activités informatiques. Faut-il voir dans l'externalisation un effet de mode ou au contraire la manifestation d'une tendance durable et un nouveau partage des tâches entre acteurs économiques ? Plus particulièrement, dans quelle mesure l'externalisation peut-elle concerner les fonctions d'études informatiques ? Quels sont les facteurs favorables à l'externalisation, quels en sont les risques et les limites ? Ce sont ces questions qui sont à l'origine de la thèse.

Les travaux s'appuient sur quatre référentiels théoriques, répondant chacun à une approche différente du problème et à un questionnement particulier. Approche stratégique, qui revient à s'interroger au travers de la théorie de la ressource sur les compétences de base de l'organisation. Approche industrielle, dans laquelle est étudié l'impact de l'évolution des processus de production et de la technologie sur les modes d'organisation. Approches de la théorie de l'agence et de la théorie des coûts de transaction, qui posent la question du contrôle de l'activité et de la relation avec le fournisseur, interne ou externe, d'applications informatiques.

L'étude empirique a porté sur 12 entreprises françaises et comporte deux volets. Une analyse quantitative, portant sur 60 projets de développement d'applications mis en oeuvre dans les deux dernières années par ces entreprises ; une analyse qualitative, dans laquelle sont identifiées les préoccupations stratégiques, industrielles et de contrôle de ces différentes organisations et l'impact sur les choix d'externalisation.

L'étude empirique permet de décrire au travers d'une typologie les pratiques actuelles. Elle permet également d'analyser différentes formes contractuelles (organisation interne, régie, forfait) et d'associer à ces formes contractuelles différents facteurs de contingence. Sur un plan stratégique, notre recherche montre que les entreprises étudiées envisagent volontiers l'externalisation des étapes techniques du processus, y trouvant en particulier des avantages en termes de réactivité ; elles tiennent en revanche à maintenir en interne la compétence nécessaire à l'analyse des besoins et à l'identification des opportunités liées aux nouvelles technologies informatiques. Sur un plan industriel, l'irrégularité croissante des flux de production et l'hétérogénéité de la technologie sont identifiés comme des facteurs jouant en faveur d'une spécialisation des acteurs. En revanche, la principale limite à l'externalisation des études informatiques s'avère être le problème du contrôle dans un marché caractérisé par une forte incertitude *a priori* sur la qualité des productions. Cette dernière constatation nous amène ainsi à nous interroger sur la mise en oeuvre de contrôles efficaces, tant en interne qu'en externe et sur le rôle des différents acteurs, notamment de la direction informatique, dans ces processus de contrôle. A cet égard, la pratique d'une externalisation partielle, réfléchie et maîtrisée apparaît comme une opportunité pour les organisations et comme un mode d'apprentissage vers une gestion plus efficace de l'activité.

Introduction

"La révolution informatique envahit tous les aspects de notre économie sans qu'aucune entreprise puisse y échapper. La technologie de l'information modifie le fonctionnement de l'entreprise et influence la totalité du processus de création des produits. L'importance de la technologie de l'information n'est pas en question. Il s'agit de savoir non pas si elle aura des effets sensibles sur la position des entreprises vis-à-vis de leurs concurrents, mais plutôt de quelle manière et à quel moment ces effets se feront sentir. Les entreprises qui auront su prévoir l'efficacité de la technologie informatique contrôleront les événements."

Le message émis par M.E. Porter en 1985, dans un article publié dans la *Harvard Business Review*, est clair et impératif : le système d'information et les nouvelles technologies de l'information peuvent avoir, ont déjà ou auront bientôt un impact stratégique majeur sur les différentes activités des entreprises. Notre société moderne est dominée par l'information. Information indispensable, que ce soit dans la conception des produits, les processus de production, la distribution et la vente, les techniques de segmentation de clientèle. Chaque production de bien ou de service génère en parallèle une production d'information à destination de l'entreprise, de ses partenaires ou de ses clients. L'échange des biens et services sur les marchés s'accompagne d'un flot toujours croissant d'informations manipulées et échangées entre les différents acteurs. Information omniprésente, qui va prendre une place plus importante encore, poussée par une évolution technologique incontournable dont le multimédia, les autoroutes électroniques chères à l'administration du Président Clinton et les entreprises virtuelles sont des manifestations hier jugées comme futuristes mais qui seront bientôt habituelles.

L'article que M.E. Porter consacre à ce sujet, en collaboration avec un associé du cabinet Arthur Andersen, V.E. Millar, initie un nombre impressionnant de travaux de recherches et d'opérations de conseil que beaucoup d'entreprises achètent, ce avec d'autant plus d'intérêt qu'un nombre croissant d'entre elles se reconnaît dans les activités à "haute intensité informationnelle" décrites par l'auteur. Vers la fin des années 80, on ne parle donc que d'impact stratégique, de nouvelles formes de marché, de réseaux, de CIM (*computer integrated manufacturing*), de systèmes d'information marketing ... et de nouvelles technologies.

La crise économique qui apparaît avec la guerre du Golfe va mettre quelque frein à cet entrain ; elle va surtout reposer le problème de la gestion et du coût des systèmes d'information. Pour les entreprises se pose concrètement la question de la conduite de cette évolution vers une société informationnelle et de la mise en oeuvre des moyens nécessaires pour structurer, collecter, stocker, échanger et utiliser l'information. Comment gérer la ressource "système d'information" ? Quels moyens y consacrer ? Avec quels espoirs de valeur ajoutée ? Comment limiter les coûts sans sacrifier les objectifs à atteindre ?

Le problème n'est en soi pas nouveau ; les organisations sont confrontées à ces questions depuis les débuts de l'informatisation. La complexité de la technologie, l'ampleur des problèmes à traiter, l'importance des investissements engagés ont dans un premier temps conduit les grandes entreprises à créer en leur sein une classe d'experts, la direction informatique, et à lui confier la responsabilité de la ressource. Au début des années 1980, la direction informatique devient, aux côtés de la direction financière et des directions opérationnelles, une direction importante des grandes entreprises. La politique de centralisation interne lui confère un pouvoir de contrôle sur la ressource informatique : les achats de matériel passent obligatoirement par elle, la très grande majorité des logiciels sont produits sur mesure sous la responsabilité des équipes internes.

C'est alors qu'intervient ce qui fait l'effet d'une "bombe" dans le milieu des systèmes d'information : en 1987, Eastman Kodak décide d'externaliser une partie de ses activités informatiques, lançant ainsi un débat qui agite la communauté des systèmes d'information, sous le nom d'*outsourcing*. C'est la première fois qu'une grande entreprise décide de façon aussi spectaculaire de rompre avec le choix de la ressource interne et d'externaliser une partie de ses activités informatiques. Ce premier contrat va être suivi de beaucoup d'autres, dans différents secteurs économiques, y compris des secteurs où l'information est stratégique, comme la banque ou l'assurance. L'appel à la ressource extérieure concerne dans un premier temps surtout les activités d'exploitation informatique, la gestion du parc machine et les

télécommunications, mais certaines entreprises envisagent d'aller plus loin et d'externaliser également la partie développement d'application.

Chacun sait que les grandes entreprises françaises font en partie appel à la sous-traitance auprès de Sociétés de Services et d'Ingénierie en Informatique pour les aider à développer leur portefeuille d'applications. Mais quelle forme prend la sous-traitance ? Quels sont les facteurs qui conduisent à retenir un choix particulier de ressources ? Quel rôle l'informatique interne joue-t-elle dans le domaine du développement d'applications et quelles évolutions peut-on anticiper ? Plus radicalement, peut-on envisager d'externaliser complètement cette fonction ? Ce sont ces questions qui sont à l'origine de ce travail.

Le problème central de cette recherche est l'externalisation du développement d'applications informatiques.

Une réflexion sur l'externalisation du développement d'applications nous a semblé particulièrement intéressante pour trois raisons :

- Les enjeux liés au système d'information sont majeurs pour la plupart des entreprises : non seulement les activités opérationnelles et le contrôle de ces activités sont dépendants de son bon fonctionnement, mais de plus en plus le système d'information joue un rôle important dans l'organisation des marchés et les nouvelles stratégies, qu'elles soient de différenciation ou d'alliance. Dans ce contexte, il est vital pour l'entreprise de s'assurer de la bonne maîtrise de son système d'information.

- Alors que les services liés au matériel sont relativement banalisés, la définition et la mise en oeuvre des applications informatiques reste étroitement dépendante de l'activité et du contexte spécifique de chaque organisation. Le développement d'applications est ainsi sans doute l'activité informatique la plus difficile à externaliser.

- Le rôle de la sous-traitance, les critères qui poussent les entreprises à y faire appel, les limites à ce qui peut être confié à un prestataire extérieur dans un projet de développement applicatif ont été jusqu'à présent très peu étudiés dans la littérature des systèmes d'information.

Au moment où l'offre des professionnels du secteur informatique se structure pour proposer des solutions de plus en plus complètes et où des solutions radicales d'externalisation peuvent ainsi être envisagées, ce manque de connaissance n'est plus acceptable.

L'objectif de ce travail est d'**analyser et de comprendre les arbitrages** entre ressource interne et externe dans le domaine du développement d'applications informatiques.

Quelques grandes orientations nous ont guidé tout au long de ce travail.

Nous savions, par une expérience professionnelle acquise pendant une dizaine d'années dans le développement et la mise en oeuvre de systèmes d'information, que notre objet d'étude, l'activité informatique et plus précisément le développement d'application, présentait des caractéristiques originales qui devaient impérativement être prises en compte dans l'analyse des choix. C'est pourquoi nous avons accordé une large part à l'analyse des composantes de la ressource de développement.

La question de l'externalisation ne se limite pas au domaine des systèmes d'information. Elle a été posée pour de nombreuses activités de l'organisation ; nous nous sommes efforcés d'aborder le problème du choix de la ressource dans une approche analytique en confrontant les théories, concepts et modèles développés dans la littérature aux spécificités de notre objet d'étude. Pour chaque approche théorique étudiée, nous avons ainsi cherché à déterminer comment elle pouvait être déclinée et quelle mesure elle pouvait s'appliquer au problème spécifique étudié.

Compte tenu du peu de travaux de recherche disponibles sur ce sujet, notre étude est exploratoire quant à l'application de ces théories au développement d'applications informatiques. C'est pourquoi nous avons privilégié dans la recherche terrain la diversité des situations étudiées, accordé une large place à la description des pratiques actuelles et cherché à élaborer une typologie des choix. Nous avons voulu également mener une première analyse des facteurs influant les décisions de ressource, susceptible d'orienter des recherches ultérieures.

Ce travail comporte trois grandes parties.

La première partie présente les "données du problème". Les deux premiers chapitres traitent des spécificités du choix de ressources étudié. Le premier chapitre aborde la problématique générale d'externalisation de l'informatique. Il retrace les évolutions du comportement des entreprises et analyse le débat actuel sur "l'outsourcing". Le chapitre 2 nous permet de situer dans ce contexte général notre domaine précis de recherche ; on y trouvera la définition et les limites de notre champ d'étude.

La seconde partie consiste en l'analyse du problème, au travers de quatre approches théoriques du choix d'externalisation que nous avons confronté à notre champ de recherche. Nous avons schématisé chaque approche pour mettre en évidence ses caractéristiques propres et choisi de retenir pour chacune d'entre elles l'angle principal sous lequel elle aborde le problème d'externalisation. L'approche industrielle est centrée sur la réduction du coût de production et sur l'optimisation du processus de production (chapitre 3) ; l'approche stratégique lie le maintien de la ressource en interne à son intérêt stratégique pour l'organisation (chapitre 4) ; les deux approches contractuelles, la théorie de l'agence et la théorie des coûts de transaction sont abordés respectivement dans les chapitres 5 et 6. Une synthèse et un premier essai d'intégration de ces approches sont présentés au travers de la construction d'un modèle de recherche dans le dernier chapitre de cette seconde partie.

La troisième partie est consacrée à la présentation de l'étude empirique et de ses résultats. Le chapitre 8 décrit les résultats d'une analyse statistique portant sur 60 projets de développement d'application menés par différentes entreprises dans les deux dernières années. Il permet d'élaborer une typologie de comportements et confirme, sur l'échantillon analysé, une partie des hypothèses émises précédemment. Le chapitre 9 est la synthèse et l'analyse qualitative des entretiens menés dans ces entreprises ; il permet d'illustrer et de compléter l'analyse théorique et les résultats de l'analyse quantitative.

Première Partie :
Les données du problème

Pourquoi les entreprises se posent-elles maintenant la question d'externaliser leur informatique ? La question se pose-t-elle différemment dans le cas du développement d'applications ?

Cette partie est composée de deux chapitres.

Depuis quelques années se sont multipliés aux Etats-Unis -puis en France- les exemples d'entreprises ayant décidé de renoncer à la gestion directe de parties importantes de leurs activités informatiques. Ce phénomène, souvent repris sous son nom d'origine (*outsourcing*), a suscité de très nombreux commentaires dans les milieux de l'informatique et du management. Faut-il voir dans l'externalisation un effet de mode ou au contraire la manifestation d'une tendance durable et un nouveau partage des tâches entre les acteurs économiques ?

Nous étudierons les solutions proposées et les premiers résultats obtenus mais aussi les difficultés, les risques et les limites de l'externalisation. Nous montrerons ainsi dans le premier chapitre que l'externalisation, au-delà des aspects conjoncturels, cherche à répondre à des problèmes structurels de gestion de l'informatique.

Après cette analyse du contexte général, le second chapitre sera consacré au problème précis de l'externalisation du développement d'applications. Cette question s'est posée une première fois dans les années 80 avec le débat sur l'utilisation des progiciels, dont nous rappellerons les principaux éléments et mettrons en évidence les limites. Nous nous intéressons pour notre part au problème d'externalisation de la **partie spécifique du développement d'applications**, dont nous préciserons les contours dans la seconde section.

Chapitre 1 :
Analyse du mouvement d'externalisation
des activités informatiques

Introduction

En 1987 s'est produite une rupture dans la gestion de la ressource informatique : l'événement qui en est le symbole est l'accord passé en 1987 aux Etats-Unis entre la société Eastman Kodak et un ensemble de prestataires informatiques (IBM, Businessland et DEC), accord selon lequel elle leur confie à long terme la propriété et l'administration de son centre de calcul. Que d'autres entreprises, bien avant, aient confié des parties de leurs activités informatiques à l'extérieur ne fait aucun doute. Mais le contrat passé par Kodak présente des caractéristiques nouvelles : ce qui frappe, c'est à la fois l'aspect radical de la décision (des pans entiers de l'activité informatique sont externalisés), les promesses fabuleuses de réduction des coûts qui sont faites (annonces de baisse des coûts de 10 à 50%), la philosophie sous-jacente ("A quoi bon gérer nous-mêmes une ressource banalisée ?", déclare le directeur des systèmes d'information de Kodak), la conviction tranquille de Kodak qui s'engage pourtant dans une voie réputée dangereuse et la médiatisation de l'accord, présenté comme *la solution* aux problèmes informatiques et *la bonne règle de gestion* de cette activité, en rupture avec quarante ans de vicissitudes dans les relations de l'entreprise et de son informatique interne.

Limitée au départ à quelques entreprises et aux activités d'exploitation, l'idée de l'externalisation des activités informatiques s'est propagée et concerne maintenant également les activités de développement d'applications. Le contexte général apparemment favorable à l'externalisation de l'informatique semble être le premier moteur de la question qui nous intéresse ici et c'est pourquoi nous lui consacrons ce premier chapitre.

Notre objectif dans ce chapitre est de faire le point sur le mouvement récent en faveur de l'externalisation des activités informatiques. Quels sont les arguments des défenseurs de l'externalisation ? Quelles sont les activités concernées ? Est-ce un phénomène temporaire ou au contraire une tendance profonde et difficilement réversible ?

Pour comprendre la genèse de la question de l'externalisation des activités informatiques, nous avons adopté une démarche en deux temps.

Tout d'abord, nous ferons un bref **rappel historique de l'évolution de l'informatique** afin de déterminer comment cette ressource s'est développée dans les organisations et comment les marchés se sont structurés ; nous ferons en conclusion de cette première section le point sur les **interrogations et les problèmes actuels**.

Dans un deuxième temps, nous analyserons la **logique des propositions actuelles d'externalisation de la fonction informatique**, montrerons comment l'externalisation peut apparaître comme *la solution* aux différents problèmes conjoncturels mais aussi structurels de gestion de l'informatique et étudierons les mises en garde adressées par certains auteurs.

Section 1 - l'histoire de la gestion de la ressource informatique dans les entreprises

L'histoire de l'informatique dans les entreprises n'a guère plus qu'une quarantaine d'années mais à l'heure actuelle son usage s'est imposé dans la quasi totalité des entreprises de l'ensemble des secteurs de l'activité économique. Comment s'est opérée cette diffusion et comment les entreprises ont-elles géré la ressource informatique dans le passé ? On peut distinguer trois grandes étapes dans cette évolution : les débuts (des années 45 au milieu des années 60), la diffusion de l'informatique "classique" centralisée (jusqu'à la fin des années 70) et la période des années 80 qui est marquée par la révolution de la micro-informatique. Nous allons retracer ici brièvement les caractéristiques de ces différentes étapes, en nous appuyant sur les travaux de Birien (1990), Breton (1987), Desaintquentin et Sauter (1993), Lacrampe et Macquin (1993), Morley (1991) ainsi que sur le rapport de l'OCDE (1985).

I. Jusqu'au milieu des années 60 : les débuts de l'informatique

Si l'informatique est née dans les années 1940, les historiens s'accordent à faire remonter très loin ses origines : pour Birien, les boules et jetons de l'écriture cunéiforme des Sumériens marquent la naissance du "traitement rationnel de l'information", et intègrent déjà "les fonctions classiques de notre époque - mémorisation (les boules et les jetons), entrée (confection de la boule et des jetons), traitement (casse de la boule ou tri des jetons percés), sortie (interprétation des jetons), contrôle (comparaison du contenu du support avec l'objet de la transaction), rétroaction (évolution progressive des surfaces des boules)" (1990, p.13). Dans son histoire de l'informatique, Philippe Breton (1987) fait remonter les racines de la discipline aux techniques de l'automatisme et au mécanisme de la clepsydre inventé par le grec Ctésibios au III^e siècle avant Jésus-Christ. On retrouve dans ces deux filiations la double origine du mot informatique, amalgame de information et automatique¹. L'informatique résulte ainsi de la

¹. Mot créé en France en 1962, qui n'a pas d'équivalent dans les pays anglo-saxons où l'on parle de "*data processing*".

rencontre entre des concepts philosophiques et mathématiques (logique, codification, numération, algorithmie..) et une technique extrêmement sophistiquée : les premiers informaticiens sont, selon le terme de Breton, des "mathématiciens-programmeurs".

Le premier calculateur entièrement électronique, l'*ENIAC*², voit le jour il y a une cinquantaine d'années à l'université de Pennsylvanie. Il a été conçu à la demande de l'*US Army*, occupe 140 m², pèse plus de 30 tonnes et peut effectuer 300 opérations à la seconde. Les quelques années de l'après-guerre voient se succéder une série de découvertes décisives : le mathématicien von Neumann développe le concept de programme enregistré, Shannon élabore la théorie de l'information, Wiener publie (1948) *Cybernetics, or control and communication in the man and the machine*, qui marque le début du développement de la cybernétique.

Selon les termes de Morley (1991), on est encore à "l'ère des pionniers". Le marché se structure autour du matériel, les constructeurs s'organisent et on voit apparaître de nouveaux géants de l'industrie (IBM, Digital, Control Data). Le marché des logiciels est encore très inorganisé : IBM est à l'origine de la création de Fortran et de PL1, les langages LISP et BASIC sont inventés par des mathématiciens d'universités, COBOL est dû à l'initiative de la "Conférence On Data System Languages" (*CODASYL*) initiée par le Pentagone en 1959.

Les progrès de l'informatique sont très rapides ; elle sort très vite des laboratoires de recherche et les applications commencent à diffuser dans les entreprises. En 1951, Eckert et Mauchly commercialisent pour la première fois un ordinateur à programme intégré, l'*UNIVAC*³. Dès 1954 apparaissent les traitements informatiques effectuant le calcul de la paie et de la comptabilité. Vers la fin de cette période sont réalisés de nouveaux progrès qui permettent la généralisation de l'usage de l'informatique dans les entreprises : d'une part apparaissent les transistors au silicium, plus accessibles financièrement, et d'autre part sont développés les premiers langages universels (Fortran, 1957, Cobol, 1960).

². "Electronic Numerical Integrator and Calculator", mis en service en 1946.

³. "Universal Automatic Computer"

La technique reste toutefois encore expérimentale ; les informaticiens recrutés par les entreprises dans ces débuts de l'informatique présentent de nombreuses caractéristiques communes avec l'archétype du chercheur - connaissances approfondies, apprentissage constant, mais aussi un isolement dans la tâche rompu seulement par des contacts avec les pairs :

"L'informatisation ne touche guère que les grosses entreprises... Les informaticiens sont des techniciens, en général de formation supérieure. C'est une population très motivée, autonome, avec de grandes capacités d'adaptation et d'apprentissage. La communauté des informaticiens franchit les frontières des entreprises, souvent par le biais des constructeurs. Mais le travail de production s'effectue en solitaire : l'écriture d'un programme est une oeuvre individuelle, une création personnelle." (Morley, 1991, p. 29-31).

Il faut enfin noter enfin le climat très spécifique qui caractérise l'attitude du grand public face à l'informatique dans cette période. Née juste après la guerre et issue de programmes de recherche financés par des budgets militaires, l'informatique est dès son origine symboliquement identifiée dans l'inconscient collectif à un processus technologique, scientifique, rationnel, mais aussi étranger, dominateur ... et inéluctable. D'une part la foi dans la bonne utilisation du progrès scientifique a été ébranlée avec l'utilisation du nucléaire pendant la guerre et, d'autre part, on peut craindre que l'informatique soit l'outil centralisateur mettant en danger les libertés individuelles. Ces différentes caractéristiques sont diffusées dans le monde entier par la littérature de science-fiction, dont toute une branche s'attache à décrire la vision, généralement très pessimiste, d'un monde dominé par la technologie informatique. Le fantasme de l'informatique dominante, s'exerçant dans un monde déshumanisé se traduit ainsi largement dans la littérature populaire et engendre en réaction des craintes et des phénomènes de rejet⁴. Le jugement exprimé envers l'informatique est ambigu, traduisant la fascination exercée par le progrès technique, toujours mêlée de la crainte devant un futur inconnu, qui ne serait plus maîtrisable directement par l'homme. Il faut donc protéger l'homme et lui

⁴. Le roman de G. Orwell, "1984" a pour une génération entière symbolisé la crainte d'un pouvoir tyrannique déçu par l'usage de l'informatique.

subordonner "par construction" la machine, comme l'illustrent par exemple les "trois lois de la robotique" élaborées par Isaac Asimov⁵ dans son très célèbre "Cycle des robots".

II. Jusqu'à la fin des années 70 : diffusion de l'informatique, organisation du marché du matériel, utilisation de la ressource interne pour le logiciel

La période 1965 - 1980 est riche en inventions techniques : arrivée du microprocesseur et, par voie de conséquence, des mini-ordinateurs ; définition de la programmation structurée avec Pascal et le langage C ; invention de la "souris", qui permet à l'utilisateur de pointer et de sélectionner directement des objets à l'écran, par Engelbart dès 1968 ; démarrage du développement du système d'exploitation multi-tâches et multi-utilisateurs Unix ; débuts de l'intelligence artificielle ; arrivée du premier micro-ordinateur à connaître un succès commercial, l'Apple II, en 1977. Quelques uns de ces progrès techniques vont rapidement diffuser dans les entreprises (les mini-ordinateurs, la programmation structurée) mais cette période est surtout celle où se prépare la révolution micro-informatique des années 80.

Les budgets informatiques des entreprises croissent régulièrement pendant toute la période. Les marchés s'organisent autour du **matériel** qui est la préoccupation dominante de ces années. IBM devient l'exemple même de la réussite et de l'organisation toute puissante. Les choix principaux en matière d'investissement informatique touchent au type de matériel utilisé pour le site central, à la capacité machine ; les responsables informatiques se définissent comme "pro-IBM" (ou "pro-DEC", ou "pro-HP"...). Dans ces choix de matériel, le discours technique des

⁵ Il y a peu de rapports réels entre le monde de la science et celui de la science-fiction. Isaac Asimov, docteur en biochimie est une exception à cette règle et ses écrits montrent un goût indubitable pour la logique et la rigueur scientifique. Les trois lois de la robotique, qui sont les postulats qui structurent le monde des robots d'Isaac Asimov sont : 1. Un robot ne peut nuire à un être humain ni laisser sans assistance un être humain en danger. 2. Un robot doit obéir aux ordres qui lui sont donnés par les êtres humains, sauf quand ces ordres sont en contradiction avec la première loi. 3. Un robot doit protéger sa propre existence tant que cette protection n'est pas en contradiction avec la première ou la deuxième loi. Les derniers écrits de ce cycle expriment la crainte d'un déterminisme technique et le souhait d'un retrait : "Tout acte d'assistance d'un robot affecte l'humanité. Les humains perdraient à la longue leurs capacités d'initiative.. L'utilisation des robots étrangle le processus d'évolution humain" (La cité des robots d'Isaac Asimov : Humanité, livre 5, 1990).

informaticiens domine sans partage, les gestionnaires s'avouant tout à la fois dépassés et rebutés par l'hermétisme des sujets traités.

Compte tenu des barrières formées par les systèmes propriétaires⁶, le marché des **logiciels** se limite principalement aux logiciels de base et aux langages de développement. Des logiciels d'application standard sont développés pour les fonctions de base de la gestion (comptabilité, paie). Les entreprises constituent à cette époque leur "stock" d'applications spécifiques développées en interne en COBOL sur gros systèmes. La productivité est faible et les cycles de développement longs.

C'est à cette période qu'apparaît le marché des services. Il faut noter ici la spécificité des SSII (Sociétés de Services et d'Ingénierie Informatique) françaises qui se développent dans les années 70 sous l'impulsion de grandes banques et entreprises⁷. Les SSII se spécialisent dans la vente de temps d'informaticiens pour la programmation de logiciels spécifiques ; leur clientèle est formée de très grandes entreprises publiques et privées ; elles s'appuient sur des méthodologies très formalisées comme Merise et sur des outils logiciels (Sauviat, 1991).

Du milieu des années 60 à la fin de la décennie 70, l'utilisation de l'informatique se généralise. La complexité de la technologie, l'ampleur des problèmes à traiter, l'importance des investissements engagés conduisent les grandes entreprises à créer en leur sein une classe d'experts, la **direction informatique**, et à lui confier la responsabilité de la ressource.

Celle-ci assume son rôle avec plus ou moins de succès, en faisant éventuellement appel à la sous-traitance. Cet appel reste marginal : en 1979, le personnel informatique est en France globalement concentré dans les firmes utilisatrices avec 73% des effectifs, contre 15% chez les constructeurs et seulement 12% dans les SSII⁸. La direction informatique devient, aux côtés de

⁶. On oppose les systèmes "propriétaires", c'est-à-dire fonctionnant exclusivement sur le matériel d'un constructeur (voire spécifiquement sur une machine de la gamme) aux systèmes "ouverts", fonctionnant sur différents types de machines, qui vont se développer dans les années 80 et qui permettent de s'affranchir de la dépendance vis-à-vis des constructeurs.

⁷. Création de Sligos (Crédit Lyonnais) en 1970, SG2 (Société Générale) en 1972, CISI (CGE) en 1971, CISI (filialisation de ses activités informatiques par le CEA) en 1972.

⁸. Source OCDE, 1985

la direction financière et des directions opérationnelles, une des directions dominantes des grandes entreprises.

Bénéficiant du développement de l'informatique et de son image, les directions informatiques détiennent un pouvoir fort au sein des organisations. Les informaticiens disposent d'une légitimité technique considérable qui est leur atout majeur. Mieux que tout autre groupe au sein de l'organisation, ils incarnent les nouveaux modes de gestion, la rationalité dominante et le management scientifique⁹ :

"En quelques années, le paysage va être fortement modifié. La fonction informatique constitue maintenant un service à part entière, rattaché à la direction générale. L'informatisation s'étend, de façon souvent conquérante. Le service informatique dispose d'un pouvoir croissant; il représente l'avenir, la technologie avancée, la rationalisation, le progrès" (Morley, 1991).

Cependant au moment même où l'informatique apparaît triomphante, elle doit déjà faire face tant à des critiques conceptuelles qu'à des difficultés de mise en oeuvre pratique.

La critique conceptuelle est présente quasiment dès l'origine comme en témoigne l'article devenu célèbre de R.L. Ackoff, Management MISinformation Systems, (1967). Ackoff remet en cause aussi bien l'utilité des systèmes d'information que la capacité cognitive des individus à les utiliser. Dans une série de travaux passionnants, G.A. Miller (1956 et surtout 1967) étudie les processus de mémorisation et de recodification de l'information et identifie les "*chunks*", unités d'information construites par chaque individu en fonction de son expérience et de ses aptitudes. Miller montre expérimentalement que la capacité d'assimilation de la mémoire à court terme est extrêmement réduite, confirmant ainsi l'une des idées à la base du concept de rationalité limitée développé par H. Simon, à savoir que l'être humain est par nature un faible

⁹. Voir par exemple l'ouvrage de H.A. Simon et J. Lesourne, "Le nouveau management : la décision par les ordinateurs" (1980).

processeur d'information. Le concept d'*information overload* connaît par la suite un développement considérable. H. Mintzberg va développer ces critiques et expliquer les raisons de la réticence des cadres pour les systèmes informatiques (Mintzberg, 1984).

Sur le plan pratique, les difficultés restent élevées. L'informatique a certes apporté des résultats, mais surtout dans l'automatisation de décisions programmées¹⁰. De plus, ces résultats ont été acquis à un coût très élevé et continu, sans que l'on puisse entrevoir une nette décreue des dépenses à court ou moyen terme. Autre difficulté, le groupe des informaticiens reste isolé au sein de l'organisation et on assiste à un phénomène de "caste" : les informaticiens se reconnaissent entre eux, communiquent peu avec les autres, ont leurs propres centres d'intérêts et objectifs visiblement différents de ceux de l'entreprise, bref ont une logique de métier plus que d'entreprise. Leur isolement est d'autant plus critiqué que les premiers résultats ne sont pas aussi spectaculaires que prévu. Leur légitimité commence à être ébranlée :

"Le service informatique se différencie du reste de l'entreprise : les informaticiens, et particulièrement ceux qui disposent d'une compétence pointue, se rattachent davantage à une communauté externe. La mobilité d'une entreprise à l'autre est forte. La culture technicienne domine et nombre d'informaticiens cherchent dans leur activité l'intérêt technique, l'expérimentation de nouveaux logiciels de base, la construction d'applications sophistiquées.. L'isolement du département informatique et les craintes de bouleversements et réorganisations que suscite l'informatisation engendrent des situations conflictuelles. La réaction des gestionnaires se traduit par un retrait, un désengagement, une sorte de résistance passive visant à gagner du temps et ralentir le mouvement d'informatisation... Par ailleurs, en quelques années, deux soucis sont exprimés : la productivité et la qualité des développements informatiques." (Morley, 1991, p.32).

¹⁰. Selon la typologie de H.A. SIMON. (1958). Les décisions programmées répondent à des problèmes structurés, habituels, pré-réglés (l'entreprise a une règle de conduite définie) et stables. Les décisions non programmées s'attaquent à des problèmes non structurés, inhabituels et pour lesquels il n'existe pas de méthode définie de résolution.

L'idée d'un contrôle des coûts et des objectifs du service informatique se développe (Strassmann, 1976, Nolan¹¹, 1977). Les entreprises éprouvent de très grandes difficultés à contrôler leurs départements informatiques :

"Arrivé au tournant des années 80, le département informatique occupait une place infrastructurelle décisive... un aspect original de cette place est qu'elle coexistait, dans la plupart des cas, avec une autonomie remarquable du département par rapport aux stratégies et aux contraintes générales de l'entreprise : il définissait lui-même ses objectifs, leurs moyens de réalisation et leurs coûts induits, sous un pouvoir de contrôle effectif assez limité de la hiérarchie" (OCDE, 1985, p.52).

III. Les années 80 : révolution de la micro-informatique, structuration de l'industrie du logiciel, extension de la gamme des services

La miniaturisation des composants est la rupture technologique qui va bouleverser ce paysage. Dans son sillon apparaissent la micro-informatique, les tableurs, les systèmes d'exploitation ouverts de type UNIX, le *down-sizing*¹²... Le dialogue homme-machine fait des progrès considérables avec la micro-informatique et la convivialité qui l'accompagne. L'image de l'informatique renoue avec l'innovation et la créativité des premiers "mathématiciens-chercheurs" mais subit un glissement significatif : on parle maintenant des "petits génies du bricolage travaillant jour et nuit dans leurs garages pour faire surgir les nouvelles sources de la convivialité" (Breton, p.215).

Le coût du matériel baisse considérablement. Le micro-ordinateur devient un produit banalisé, on entre dans l'ère de l'informatique individuelle. La diffusion de micro-ordinateurs

¹¹. On ne saurait trop recommander la lecture de l'excellent article de Nolan, 1977, ne serait-ce que pour la présentation de la discussion entre le directeur du marketing et le coordinateur informatique sur le système de facturation interne de l'informatique. Le plaisir de la lecture vient du comique d'une situation proche de l'absurde (demander à un directeur du marketing quelle partie du coût il veut réduire et lui donner le choix entre le temps CPU, le temps de connexion, les kilobytes, ou les EXCP -I/O), qui se double, pour le lecteur ayant eu professionnellement affaire à l'informatique, d'un comique de répétition, la scène ayant déjà été vécue... l'ensemble n'ayant finalement pas tellement vieilli.

¹². Evolution en faveur de "petites machines", passage de l'informatique centrale (gros ordinateurs) à l'informatique départementale (mini-ordinateurs) et personnelle (micro-ordinateurs) (Lacrampe et Macquin, 1993).

aux capacités de stockage et de traitement impressionnantes remet en cause l'architecture traditionnelle "maître-esclave" ; le nouveau modèle est celui du "client-serveur" et de l'informatique distribuée¹³ avec un développement spectaculaire des réseaux.

La décennie 80 est celle qui voit s'inverser le rapport de domination entre les constructeurs et les éditeurs de logiciels. Le marché du matériel change radicalement au profit de la micro-informatique, les coûts chutent de façon spectaculaire, provoquant une crise grave chez l'ensemble des constructeurs. Du même coup, la compétition s'intensifie sur le marché des logiciels et services, davantage porteur. **L'offre en matière de logiciels et de services se structure, la gamme des prestations s'enrichit et s'affine.** Ces tendances se sont affirmées dans les toutes dernières années.

On prend l'habitude de ne classer dans le marché du logiciel que les progiciels¹⁴ (les logiciels spécifiques développés sur mesure pour un client appartenant à la catégorie "services"). Le progiciel a été défini par l'arrêté du Ministère de l'Industrie et de l'Education Nationale du 22/12/81 comme un "ensemble complet et documenté de programmes conçu pour être fourni à plusieurs utilisateurs, en vue d'une même application ou d'une même fonction". Le CXP¹⁵ précise "qu'un usager peut utiliser [le progiciel] de façon autonome après une mise en place et une formation limitées". Notons de plus que l'utilisateur d'un progiciel paie généralement une licence non exclusive et n'achète pas un droit de propriété.

Parmi les progiciels, on peut distinguer, d'après le classement du SYNTEC-Informatique¹⁶ :

- les progiciels système, (programmes nécessaires au fonctionnement des ordinateurs),

¹³. Dans le modèle "maître-esclave", l'ordinateur central, appelé "maître", gère et effectue la totalité des travaux. Le poste de travail n'est qu'un appareil terminal, "esclave" dénué d'intelligence, destiné seulement à mettre à la portée de l'utilisateur deux outils indispensables : un clavier pour formuler ses demandes et un écran pour lire les résultats élaborés par le central. Le modèle "client-serveur", à la base de l'informatique distribuée, correspond à un partage des tâches entre le "serveur"(poste central) qui assure le stockage de données communes et les clients (postes décentralisés) capables de gérer leurs propres applications. On trouvera une présentation très claire de ces architectures dans l'ouvrage de Desaintquentin et Sauter, (1993, p.34-53).

¹⁴. Vocabulaire formé par l'association de "produit" et "logiciel", marquant ainsi le fait que le progiciel est une production faisant l'objet de transactions sur un marché.

¹⁵. Centre d'expérimentation des Progiciels, qui publie des listes détaillées des progiciels par fonction et réalise des études sur les différents produits.

¹⁶. Le Syntec-Informatique est la chambre syndicale des sociétés d'études et de conseil informatique.

- les progiciels outils, orientés vers la productivité des programmeurs,
- les progiciels applicatifs, tournés vers la satisfaction des besoins opérationnels des entreprises (horizontaux pour une fonction -comptabilité, paie etc..., verticaux pour un secteur économique -assurance, banque). Il faut également distinguer dans cette catégorie les progiciels bureautique fonctionnant sur micro-ordinateurs et vendus en grande série par de très grandes sociétés comme Microsoft, Lotus, Borland et Oracle.

Le marché des services

Les prestations sur ce marché évoluent fortement pendant la période. L'étude Eurostaf (1989) sur les SSII compare les prestations au début des années 80 et dix ans après :

- début 80, les activités s'articulent autour de trois grands axes - études et conseil, analyse-programmation (délégation de personnel) et traitement à façon / prestations machines ; les SSII sont en retrait derrière les constructeurs qui "tiennent" leurs clients par les systèmes propriétaires et derrière les clients eux-mêmes qui gardent en interne la responsabilité des études et de l'exploitation,

- dix ans après, l'ingénierie de systèmes (livraison de systèmes clé en main) s'est installée¹⁷ ; les prestations autour de progiciels se développent, les "prestations machines" s'enrichissent des "services à valeur ajoutée" (bases et banques de données, vidéotex partagé, applications monétiques) ; les SSII proposent la maintenance globale de système et la gérance d'exploitation.

A l'heure actuelle, le fait le plus marquant sur ce marché des services est l'intérêt des clients pour les prestations globales, correspondant à une prise en charge complète par le fournisseur, généralement associées à un engagement de résultats :

- le *Facilities Management* correspond à "la prise en charge partielle ou totale de la fonction informatique de l'entreprise ; il constitue à reprendre tout ou partie des ressources

¹⁷. "Les SSII profitent de la montée de la "télé-informatique" et du fait qu'à l'époque, ni les constructeurs, ni les utilisateurs n'avaient de compétences particulières dans le développement des réseaux. Elles se dotent de compétences et acquièrent ainsi la maîtrise d'oeuvre de systèmes d'envergure" (Eurostaf, 1989, p.6).

informatiques (ordinateurs, logiciels, équipes, locaux d'une entreprise) pour assurer la gestion de l'activité informatique correspondante pendant une certaine durée" (définition du Syntec),

- l'intégration de systèmes implique "la capacité, pour une SSII, maître d'oeuvre, de concevoir et de réaliser un système complet avec des éléments hétérogènes provenant de fournisseurs différents, à partir des spécifications générales fournies par le maître d'oeuvre. L'intégration de systèmes se distingue de l'ingénierie de systèmes par le fort degré d'hétérogénéité des éléments mis en oeuvre et par la taille importante du projet" (définition du Syntec).

La figure 1.1 présente la composition de l'offre des SSII en logiciels et services en 1993.

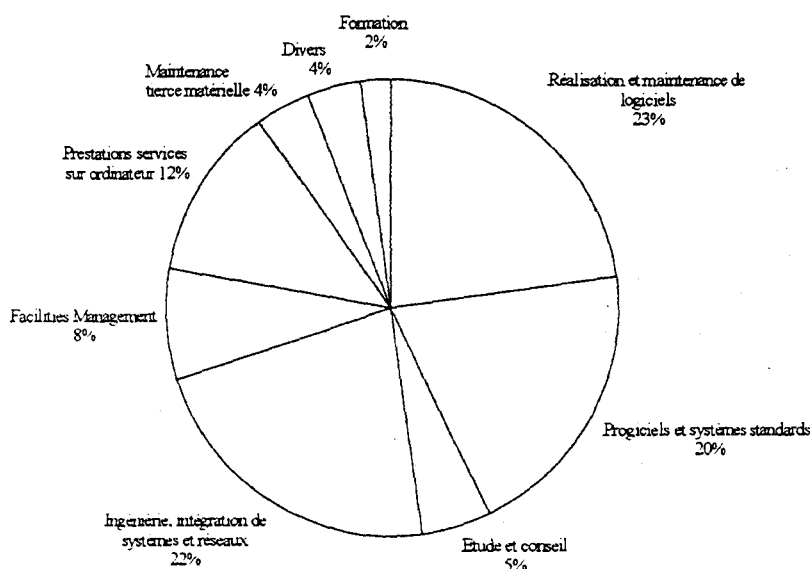


Figure 1.1. Les prestations des SSII françaises en 1993 (source Syntec, cité par O1 Informatique, 08/04/1994)

Le marché informatique français est évalué en 1993 à 225 milliards de francs, dont 47,6 % pour le matériel (source IBM, cité par Lacrampe et Macquin, p.16). Mais alors que l'on était habitué depuis de nombreuses années à un taux de croissance à deux chiffres de la dépense informatique, celui-ci a brusquement chuté depuis 1991, le secteur informatique étant ainsi

pour la première fois touché par la crise économique qui l'avait jusqu'à présent épargné (fig. 1.2).

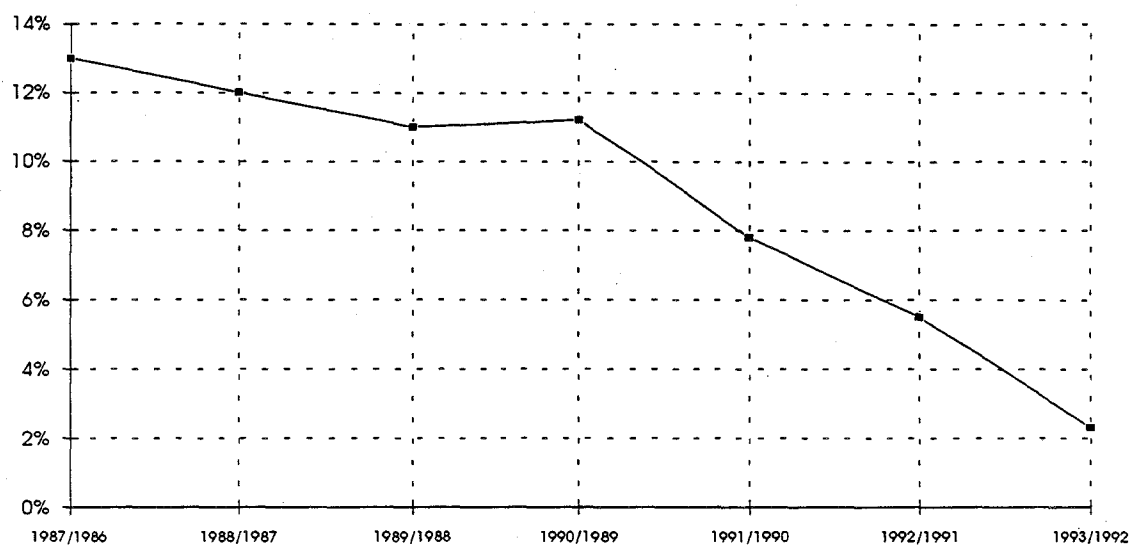


Figure 1.2. Evolution du taux de croissance
des dépenses informatiques françaises (source PAC)

La tendance devrait se confirmer en 1994, comme en témoignent les prévisions faites par différents observateurs : "taux de croissance global des dépenses très faible (entre 0 et 2%), reprise éventuelle au mieux en 1995" (01 Informatique, 01/04/94) ; "en 1994, sept directeurs informatiques sur dix devront vivre avec un budget en progression nulle, sinon en régression significative" (Le Monde Informatique, 22/04/94, d'après une enquête IDC).

Dans les entreprises, cette période est surtout marquée par la diffusion de la micro-informatique. Le PC (*personal computer*) devient "l'outil moderne du cadre", même s'il est le plus souvent très sous-utilisé, les utilisateurs se limitant à l'usage du tableur ou du traitement de texte. Les experts annoncent l'arrivée de la "génération Nintendo", préparée dès l'enfance par l'usage des consoles de jeux à une utilisation intuitive et "naturelle" des interfaces graphiques et des applications sur micro-informatique.

L'informatique de "production" (c'est-à-dire celle qui sert de support aux activités courantes) se transforme, mais beaucoup trop lentement aux yeux des utilisateurs. Ceux-ci

souhaitent des systèmes flexibles, permettant une meilleure réactivité, mais les entreprises sont handicapées dans leur évolution par le poids de la maintenance des systèmes développés à l'époque précédente, qui ont représenté des investissements considérables et que l'on hésite à remplacer : "N'oublions pas que 90% des applications informatiques aujourd'hui opérationnelles sont conformes au modèle maître-esclave. La maintenance de ces applications, nombreuses et stratégiques, occupe les équipes informatiques à plus de 60% de leur temps. Elles ne peuvent se consacrer à satisfaire comme il conviendrait les nouveaux besoins exprimés par les utilisateurs" (Desaintquentin et Sauteur, p.39). De nouveaux enjeux apparaissent dans le domaine de la communication inter-entreprises. L'échange de données informatisées (EDI) se généralise au sein des filières industrielles et commerciales (Gallia pour l'automobile, Allegro dans la grande distribution, SWIFT et Etebac pour la banque).

La responsabilité de la gestion de la ressource informatique continue à être assurée dans les années 80 principalement par la direction informatique, mais celle-ci éprouve des difficultés dans ses relations avec des utilisateurs qui critiquent son autorité, veulent pouvoir contrôler ses actions et souhaitent influencer sur les choix de ressources :

"Les relations de l'informatique avec les directions utilisatrices sont souvent tendues. On perçoit un rejet de la domination technicienne. Les directions informatiques vont être progressivement rebaptisées "direction des systèmes d'information" ou "direction de l'organisation et de l'informatique", ce qui peut être considéré comme une amorce de dialogue avec les services utilisateurs. L'interdépendance entre système d'information, système de gestion et système d'organisation a mis en lumière l'incontournabilité de l'élément humain. Il apparaît de plus en plus indispensable d'explicitier, de préciser, de hiérarchiser les objectifs affectés au système d'information automatisé, d'où nécessaire intervention des acteurs décideurs. Des expériences d'informatique décentralisée, gérée par les services utilisateurs, sont tentées. Nous sommes maintenant dans l'ère de la stratégie, de la décision." (Morley, 1991, p. 80 et s.).

Les entreprises souhaitent avoir une meilleure visibilité et un meilleur contrôle des coûts informatiques, ce qui les conduit à mettre en place des systèmes de planification informatique

et d'évaluation *a priori* des dépenses ainsi que des procédures formelles d'accord (contrats internes) entre les divisions opérationnelles clientes et la direction informatique lors du démarrage des nouveaux projets. L'idée d'une responsabilité financière des utilisateurs sur les choix d'investissements informatiques fait son chemin. Le principe d'une refacturation interne des dépenses informatiques se généralise dans les grandes entreprises : le coût de l'informatique entre dans les budgets des utilisateurs et devient *ipso facto* une préoccupation majeure.

Conclusion

Nous avons repris dans le tableau 1.1 les principaux éléments de ce rappel sur l'histoire de l'informatique. Desaintquentin et Sauter (1993) considèrent que nous vivons actuellement un tournant de cette histoire, une "fracture dans l'ordre établi". Les références techniques (architecture centralisée), culturelles (respect de l'expertise scientifique et technique des informaticiens) et organisationnelles (pouvoir des directions informatiques) sont ébranlées dans un schéma qui se dégage plus de dix ans après l'apparition de la micro-informatique dont il est issu.

	1945 - 1965	1965 - 1979	Années 80	Aujourd'hui
Tendances techniques	Ere des pionniers	Informatique centralisée	Révolution de la microinformatique	Mise en place des architectures client-serveur
Diffusion de l'informatique dans les organisations	Ne touche que quelques grandes entreprises	Grandes et moyennes entreprises	Généralisation à tous secteurs et toutes tailles d'entreprises	Ressource banalisée
Marchés	Embryonnaires	Domination du matériel (systèmes propriétaires) Développement des services	Baisse spectaculaire des prix du matériel Organisation du marché des logiciels	Situation nouvelle de crise, renouvellement de l'offre de services (prestations globales), pression de l'offre
Image de l'informatique	Ambivalence vis à vis du progrès ; craintes d'une technique dominatrice	Forte légitimité du "management rationnel" mais critiques conceptuelles et pratiques	Retour à la créativité originelle avec la micro-informatique, rejet de l'informatique classique	"Société de l'information", passage au second plan des aspects techniques
Image des informaticiens	"mathématicien - chercheur"	Expert membre d'une caste	"petit génie au fond d'un garage"	Désacralisation
Gestion de la ressource		Directions informatiques très puissantes	Directions informatiques ; montée en puissance du rôle des utilisateurs	Responsabilité première aux utilisateurs
Contrôle de l'informatique		Quasi inexistant	Refacturation du coût de l'informatique aux utilisateurs	Idem + limitation stricte des budgets informatiques
Appel à la ressource extérieure	Produits de "base": matériel, logiciels systèmes et langages de développement	Produits de "base", prestations machines et intellectuelles (la ressource reste principalement interne)	Idem + progiciels (applicatifs horizontaux et bureautique) et ingénierie de systèmes	Idem + nouvelles offres : facilities management, intégration de systèmes. Proposition de l' externalisation systématique de certaines fonctions

Tableau 1.1. Synthèse des évolutions de l'informatique

Concernant la gestion de la ressource, trois aspects principaux nous semblent à retenir.

1. On peut considérer à présent que la première grande étape de diffusion de l'informatique est terminée ; celle-ci devient une **ressource banalisée**, en particulier du fait de la micro-informatique.

2. Les entreprises ont eu dès l'origine à faire appel à des ressources extérieures ; ce recours se limitait à l'origine à la fourniture de matériel et de logiciels de base, puis s'est étendu aux services professionnels et aux progiciels. La question est maintenant de savoir s'il faut **systématiser** ce recours à l'extérieur et redéfinir un partage des tâches entre les entreprises et les opérateurs du marché informatique.

3. Le contexte de crise conduit les différents prestataires du marché informatique à mener une politique commerciale agressive, à tenter de prendre en charge la responsabilité de projets importants d'intégration de systèmes (ce qui leur permet de justifier des prix élevés) et à rechercher des prestations globales à long terme (du type contrat de facilities management) ; la "**pression de l'offre**" est forte.

4. En interne, on assiste à une triple **remise en cause des directions informatiques** :

- sur le plan **financier**, l'attention des directions générales s'est portée depuis les débuts de la crise sur les budgets informatiques avec une acuité toute nouvelle,
- de plus en plus on considère que la **responsabilité des investissements informatiques** doit être partagée avec les utilisateurs,
- enfin même sur le plan technique, leur **rôle d'expert est discuté** par des utilisateurs qui ont pour référence les avantages et la convivialité de la micro-informatique. L'informaticien interne risque d'être considéré comme "le frein à l'évolution" alors même que, compte tenu du poids des systèmes existants, la cohabitation entre l'architecture centralisée classique et les nouveaux modèles "client-serveur" durera sans doute des années et sera techniquement très délicate à opérer.

Comme le notent avec une provocation volontaire Desaintquentin et Sauter, "l'ordinateur est descendu de son piédestal, entraînant dans sa chute ceux qui naguère étaient considérés par

les utilisateurs et les responsables financiers comme les grands prêtres d'une informatique mythifiée. Une certaine suspicion peut même s'installer, à l'endroit de l'informaticien, soupçonné de n'être plus vraiment dans le coup. Coincé entre une direction générale en mal d'économie, des utilisateurs chaque jour plus exigeants et des constructeurs à la recherche d'un nouveau statut, le responsable informatique risque de vivre très mal cette situation" (1993, p.29).

Faut-il pour autant interpréter les trois époques que nous avons décrites comme un cycle "naissance, maturité, déclin" pour les directions informatiques ou, selon une version plus optimiste, comme "naissance, croissance et crise nécessaire avant le renouveau"? Desaintquentin et Sauter tranchent nettement en faveur de la seconde hypothèse, arguant que l'entreprise ne peut se passer du support de ses informaticiens pour franchir cette nouvelle étape.

La remise en cause actuelle montre en tous cas une crise profonde, qui se traduit dans le discours (parfois mis en pratique) sur l'externalisation des services informatiques, que nous allons maintenant étudier.

Section 2 - le débat sur l'externalisation de l'informatique

Les premières recherches consacrées directement aux choix d'externalisation des ressources informatiques sont apparues très récemment, quand ces choix ont été mis en pratique par quelques pionniers, ce qui a suscité une attention considérable (la décision de Kodak en 1987 d'externaliser son centre de calcul et les fonctions d'exploitation a fait l'effet d'une révolution et a largement été commentée). L'externalisation (ou l'*outsourcing*¹⁸, pour reprendre le terme généralement utilisé dans la presse professionnelle), qui étonne et inquiète au premier abord, apparaît assez vite aux yeux des observateurs comme l'une des tendances fortes du marché de la décennie 90 (Loh et Venkatraman, 1992, Martinsons, 1993).

Afin d'ordonner la discussion sur ce thème apparu si récemment que le débat est encore relativement confus, nous adopterons une démarche progressive en abordant successivement les questions suivantes :

- Externaliser quoi ? Sous le terme d'externalisation sont regroupées des prestations très différentes en termes de contenu, de modalités et d'engagement.

- Externaliser pourquoi ? Nous présenterons ici les principaux arguments des partisans de l'externalisation et chercherons à mettre en évidence la logique des choix d'externalisation.

- Quelles sont les limites de l'externalisation ? Nous ferons écho des travaux de chercheurs qui dénoncent le phénomène de mode autour de l'externalisation et la présentation exagérément optimiste qui a pu être faite de ses résultats et cherchent à en dégager les limites.

Nous commençons cette discussion par un rappel des définitions de l'externalisation proposées par différents auteurs.

¹⁸. L'informatique est, on le sait, l'un des lieux privilégiés d'invasion de la langue anglaise. La pratique de l'*outsourcing*, née aux Etats-Unis, a essaimé en France et on peut regretter que ce soit sous son appellation d'origine qu'elle ait été popularisée. Pour notre part, nous utiliserons de préférence le terme "externalisation".

I. Définitions de l'externalisation

L'externalisation est un concept général en gestion qui correspond au fait pour une entreprise de confier, de déléguer à une autre, l'exécution d'une tâche, d'une activité, plutôt que de l'assurer elle-même.

Le terme est maintenant largement utilisé dans la littérature des systèmes d'information où il désigne "l'action de sous-traiter tout ou partie des activités de systèmes d'information à un fournisseur extérieur plutôt que de s'appuyer sur des ressources et des compétences internes" (Martinsons, 1993).

Loh et Venkatraman (1992) insistent sur l'importance du recours à l'extérieur en définissant l'externalisation comme "une contribution *significative* de fournisseurs extérieurs aux ressources matérielles et/ou humaines entrant dans tout ou partie des composantes de l'infrastructure en systèmes d'information de la firme utilisatrice".

Lacity et Hirschheim (1993) donnent comme exemple de l'ampleur pouvant être atteinte par le phénomène d'externalisation le contrat passé entre Enron et EDS, portant sur 750 millions de dollars et dix années.

Eckerson met en avant le transfert de responsabilité qui s'opère par le biais de l'externalisation : "L'externalisation s'est développé récemment pour prendre en charge des systèmes multiples et représente un transfert significatif d'actifs, de droits et de personnels chez un fournisseur qui assume la responsabilité des résultats" (cité par Lacity et Hirschheim, 1993, p.3, traduit par nous).

Laigle souligne le caractère définitif de certaines formes d'externalisation : "L'outsourcing correspond à la volonté de sous-traiter *pour toujours* certaines fonctions. Il n'est plus question de réversibilité, mais on tient évidemment à garder à tout moment -en cas de drame- la possibilité de portabilité vers une autre SSII" (Dossier Facilities Management du Monde Informatique du 07/01/94, p.18).

Dans sa forme radicale, "externaliser" est donc choisir une sous-traitance structurelle, cesser complètement d'exercer certaines activités en interne, opter pour une délégation à moyen ou long terme. Lacity et Hirschheim parlent de "donner les clés du royaume" au fournisseur qui "fait fonctionner, gère et contrôle les fonctions du système d'information".

Mais il existe également des formes atténuées d'externalisation, limitées à un domaine précis de l'activité informatique et qui correspondent à une sous-traitance de moyenne ou longue durée avec engagement sur les résultats atteints.

II. Externaliser quoi ?

L'externalisation peut être envisagée pour une partie ou pour l'ensemble ou des activités des systèmes d'information d'une organisation. Elle peut ainsi concerner (Loh et Venkatraman, 1992) :

- la gestion du parc de matériel informatique et sa maintenance,
- les activités d'exploitation du centre de calcul,
- les réseaux et télécommunications,
- l'architecture générale et la planification du système,
- l'intégration technique de systèmes,
- le traitement de données,
- la maintenance et le développement d'applications.

Une erreur fréquemment répandue est de croire que l'externalisation est une décision globale qui consisterait à confier en une seule fois toute son informatique à un partenaire extérieur (*tout* externaliser contre *tout* faire en interne). Martinsons (1993) s'élève contre cet *a priori* qui résulte d'une simplification dans la traduction de la réalité : "les managers doivent réaliser que l'externalisation n'est pas une solution binaire. Une organisation peut choisir de sous-traiter des parties spécifiques du management et des opérations de son système d'information. Elle peut sous-traiter pour une période courte ou longue, pour un ou plusieurs domaines d'activité, selon la localisation géographique, par contrat ou en vendant des actifs".

On regroupe sous l'appellation de *Facilities Management* (FM) certains contrats d'externalisation. Ceux-ci concernent en majorité des activités de production (prise en charge de l'exploitation informatique avec ou sans hébergement du matériel, gestion des réseaux, services télématiques), la gestion d'applicatifs existants et des opérations de migration.

Laigle (1994) propose une classification des contrats de Facilities Management en trois formes, en fonction de leur durée et de leurs objectifs :

Objectif	Durée du contrat		
	courte (1 à 2 ans)	longue (4 ans)	"définitif"
Support temporaire	FM "précaire" ex : maintenance d'une application devant disparaître, peu ou pas d'évolutions à gérer		
Délégation avec possibilité de reprise en interne		FM standard ex : reprise d'un ensemble de fonctions ou d'applications et gestion des évolutions par la SSII	
Abandon de la gestion en interne de fonctions banalisées			FM de type "véritable externalisation" ex : exploitation quotidienne, maintenance applicative, gestion du parc matériel

Tableau 1.2. La classification des contrats de Facilities Management (Laigle, 1994)

III. Externaliser pourquoi ? Les raisons évoquées en faveur de l'externalisation

Nous examinerons dans un premier temps deux articles parus en 1987 (l'année même où Kodak a pris la décision d'externaliser une partie de ses activités de système d'information) et qui nous semblent révélateurs quant aux interrogations qui ont favorisé l'apparition de cette

pratique, puis verrons les arguments actuels des fournisseurs et des utilisateurs de l'externalisation.

3.1 L'externalisation annoncée par Dearden et l'analyse des stratégies de systèmes d'information de Parsons et Ward

3.1.1 la thèse de Dearden : l'externalisation vue comme le facteur permettant d'améliorer la qualité et le contrôle

Dearden est bien connu dans le domaine des systèmes d'information pour ses prises de position contradictoires¹⁹. Il est, à notre connaissance, l'un des premiers à avoir anticipé le mouvement d'externalisation, notamment dans l'article paru en 1987 : "The Withering Away of the IS Organization".

Pour Dearden, l'époque des services informatiques internes est en voie d'achèvement. Selon l'auteur, les services actuels ne seront pas forcément démantelés mais exerceront leur activité à l'extérieur de l'entreprise. A l'appui de sa thèse, Dearden cite les arguments suivants :

- grâce aux économies d'échelle induites par les progiciels, le coût du logiciel va baisser considérablement,
- les sociétés de services informatiques vont se spécialiser horizontalement (par type de logiciel) et verticalement (par secteur d'activité) alors que les services internes interviennent sur tous types de travaux,
- les sociétés spécialisées en informatique sont mieux à même de gérer le personnel informatique et d'évaluer la qualité des performances,
- la pression de la concurrence garantira un excellent niveau de performance,
- la tendance actuelle de considérer les services informatiques comme des centres de profit est le premier pas vers l'externalisation.

¹⁹. L'article "MIS is a Mirage" (1972) en est un exemple : l'auteur y remettait en question les systèmes d'information pour le management.

Ce qui est nouveau dans le discours de Dearden, c'est l'apparition d'une argumentation en faveur de l'externalisation appuyée non seulement sur les coûts mais également sur la **qualité** de la production. Pourquoi la qualité serait-elle mieux assurée par un service extérieur à l'entreprise ? On retrouve ici les effets de la spécialisation industrielle, cependant d'autres facteurs de qualité sont mis en avant par l'auteur. Tout d'abord, Dearden pense que le marché permettra d'exercer un contrôle sur les performances des informaticiens, par le biais de la concurrence, alors que l'organisation interne s'avère, elle, incapable de remplir cette fonction. Ensuite l'auteur pense que cette externalisation permettra de mettre en oeuvre un processus de décentralisation et de prise de responsabilité par les utilisateurs. Les dirigeants de l'entreprise, ne pouvant plus s'adresser aux informaticiens internes, délégueront naturellement aux responsables opérationnels la charge de leurs systèmes d'information. Ceci ne peut selon Dearden que concourir à une meilleure qualité des applicatifs, les utilisateurs étant les mieux placés pour définir les priorités et pour mener une gestion proactive ou au moins réactive de leur système d'information.

Comment sera gérée cette externalisation ? Pour l'auteur, externalisation des systèmes d'information et décentralisation de la responsabilité en interne vont de pair. Dearden est un fervent partisan de structures de décision très décentralisées, qui lui semblent être la forme adéquate pour les entreprises en réaction aux contraintes actuelles d'un marché très mouvant. Il considère également que la gestion des systèmes d'information peut et doit être décentralisée car très peu d'informations sont effectivement partagées entre les différents services de l'organisation, ce qui justifie l'existence de systèmes d'information locaux et indépendants.

3.1.2 Ward ou le "marché libre" vu comme un mode de gestion approprié de l'informatique

Les travaux de Ward (1987) traitent également du sort de l'informatique interne, mais l'auteur adopte une perspective à la fois évolutive et contingente de la structure et de l'organisation des services informatiques. Citant des travaux non publiés de Parsons, Ward identifie **six stratégies possibles** concernant les systèmes d'information, qui sont :

1. mal nécessaire (*necessary evil*)

Les systèmes d'informations sont limités aux obligations légales. Seuls les projets à très fort retour sur investissement sont entrepris. Cette stratégie présente beaucoup d'inconvénients, sauf si les systèmes d'information sont marginaux par rapport à l'activité de l'organisation.

2. ressource rare (*scarce resource*)

L'informatique interne est un centre de coût. Les investissements sont contrôlés sur des critères financiers. L'efficacité est privilégiée ce qui peut diminuer l'efficacité : la gestion très stricte des systèmes d'information les empêche de jouer un rôle d'arme concurrentielle.

3. monopole (*monopoly*)

Les services internes sont la seule source autorisée. L'efficacité est mesurée uniquement par la satisfaction des utilisateurs. Cette configuration ne favorise pas l'innovation; les systèmes d'informations ne sont pas perçus comme générateurs d'avantage concurrentiel.

4. marché libre (*free market*)

Les services internes sont mis en concurrence avec les fournisseurs extérieurs. Les dirigeants laissent jouer les règles du marché. Les utilisateurs décident. Ward identifie deux risques liés à cette stratégie : une perte d'efficacité due à la redondance de certains investissements et un développement inégal des systèmes d'information dans l'organisation.

5. centralisation et planification (*centrally planned*)

Le système d'information est totalement intégré et lié à la stratégie générale de l'entreprise ; les services informatiques sont centralisés et internes. L'accent est mis sur l'identification d'opportunités stratégiques liées aux systèmes d'information. Cette politique consomme beaucoup de ressources (investissements et *temps* consacré au domaine par les dirigeants).

6. force d'attaque (*leading edge*)

Cette stratégie est retenue quand les systèmes d'information sont considérés dans toute l'organisation comme le moyen de créer un avantage concurrentiel. L'accent est mis sur la recherche et développement ; les dirigeants sont très impliqués dans le processus. La difficulté est de créer les applications efficaces qui permettront d'opérationnaliser les idées des dirigeants. Le coût de cette politique est élevé.

Pour l'auteur, le choix entre les différents modes d'organisation dépend à la fois du type de stratégie adoptée par l'organisation dans son ensemble (stratégie de coût, de différenciation ou de niche) et de la période de croissance du système d'information. Il reprend le modèle en six périodes de Nolan (1979) et identifie les politiques susceptibles d'intervenir aux différents stades de l'évolution. Les six périodes sont regroupées en deux phases principales : la "gestion de l'informatique" regroupe les trois premiers stades (initiation, contagion, contrôle), la "gestion de l'information" les trois derniers (intégration et stades suivants).

Chacun des six modes d'organisation est associé à un contexte donné dans lequel il se montre efficient (tableau 1.3).

Choix d'organisation des systèmes d'information adapté...	...selon la stratégie de l'entreprise	...selon la période de l'évolution (Nolan)	
		premiers stades (gestion de l'informatique)	stades avancés (gestion de l'information)
"mal nécessaire"	coût	uniquement au stade 1 (initiation)	
"ressource rare"	coût	adapté au dernier stade (contrôle)	
"monopole"	différenciation	adapté aux trois premiers stades (initiation, diffusion, contrôle)	
"marché libre"	coût	adapté au stade 2 (diffusion)	adapté aux stades avancés
"centralisation, planification"	différenciation		adapté aux stades avancés
"force d'attaque"	différenciation		adapté aux stades avancés

Tableau 1.3. Les choix d'organisation des systèmes d'information :

l'approche contingente de Ward

La stratégie de "marché libre" nous intéresse particulièrement ici parce qu'elle fait intervenir explicitement les fournisseurs extérieurs. Selon l'auteur, cette stratégie va permettre dans un premier temps, alors que l'entreprise est peu familiarisée avec l'informatique, d'accélérer la phase de diffusion. Mais elle peut également intervenir dans les étapes avancées du processus de croissance. Le rôle dévolu au "marché libre" apparaît alors très clairement : il s'agit de limiter les coûts en bénéficiant de l'effet de concurrence.

De façon opposée à Dearden, Ward ne voit principalement dans l'externalisation qu'un *avantage de coût* ; il considère même que celle-ci n'est pas adaptée aux stratégies de différenciation ni aux situations dans lesquelles les systèmes d'information sont générateurs d'avantage concurrentiel pour l'entreprise. Dans ces deux cas, les liens entre le système d'information et la stratégie de l'entreprise sont très forts ; les dirigeants y consacrent une part importante de leur temps, Ward considère alors que la gestion interne des systèmes d'informations est mieux adaptée, même si elle s'avère coûteuse.

Ces deux articles préfigurent bien, à notre avis, le débat actuel sur les motivations de l'externalisation.

Tout d'abord, avec Dearden, on voit apparaître, à côté de la justification économique, des motivations :

- de qualité de la prestation, (le fournisseur extérieur est un "véritable professionnel" de l'informatique, qui est son activité principale).

- et surtout de contrôle. Le point central du raisonnement de l'auteur tient dans la vision conflictuelle qu'il adopte des relations utilisateurs-informaticiens internes. L'externalisation est vue comme la solution permettant de résoudre le problème posé par une situation dominante de l'informatique interne et le moyen de rendre le pouvoir aux utilisateurs.

Ward permet quant à lui, d'envisager différents objectifs et différents modes de gestion des systèmes d'information. Le mode de gestion par le marché est identifié comme une des stratégies adaptées aux stades supérieurs d'évolution des systèmes d'information ; l'auteur considère même que le "marché libre" est la meilleure solution si l'objectif de l'entreprise est avant tout de réduire les coûts. En revanche, en s'appuyant sur une vision cette fois-ci

consensuelle des relations utilisateurs-informaticiens internes, l'auteur préfère la gestion interne dans tous les cas où les systèmes d'information sont directement liés à la création d'avantage concurrentiel pour l'entreprise.

Ces différents thèmes vont être repris dans les arguments des partisans de l'externalisation, même s'ils sont en pratique souvent éclipsés par l'argument central des fournisseurs : la réduction des coûts de l'informatique.

3.2 Les arguments des partisans de l'externalisation : avant tout la baisse des coûts

Dans le discours sur l'externalisation, la réduction des coûts apparaît comme le premier facteur mis en avant tant par les vendeurs que par les entreprises clientes pour expliquer la décision d'externalisation. Nous allons revenir sur cet argument avant de montrer comment ce débat s'inscrit de façon plus globale en résonance avec les préoccupations actuelles, telles que nous les avons décrites dans la première section de ce chapitre.

Lacity et Hirschheim (1993) ont fait une revue de la presse américaine professionnelle qui fait écho d'économies de 10% à 50% sur les budgets de systèmes d'information suite à une décision d'externalisation. Nous reprenons quelques uns des exemples cités par ces auteurs :

"Un des responsables des systèmes d'information de American Standard déclare que l'externalisation des activités d'exploitation informatique et de réseau a réduit ses coûts de 40%" (1990) ; "Le vice président de la banque Hibernia National anticipe des économies de l'ordre de 25 à 100 millions de dollars suite à son contrat d'une durée de 10 ans pris avec IBM" (1990) ; "Le directeur financier de Kendall estime que son contrat avec Kentech lui permettra d'économiser chaque année 500.000 dollars de charges salariales" (1989).

Loh et Venkatraman (1992) ont mené une analyse statistique sur 57 grandes entreprises américaines appartenant à différents secteurs d'activité et testé un ensemble d'hypothèses sur les raisons des choix d'externalisation au moyen d'une analyse en composantes principales. Deux hypothèses concernant l'importance du facteur coût sont validées, ainsi qu'une hypothèse selon laquelle la performance des systèmes d'information (mesurée avec un indicateur appuyé

sur le coût des systèmes d'information, à savoir le ratio : chiffre d'affaires total sur coût des systèmes d'information) est corrélée négativement à l'externalisation.

En France, même si les entreprises sont plus discrètes sur les montants concernés, la réduction des coûts apparaît également comme le premier objectif :

"En confiant notre informatique, nous parviendrons à un budget stable. L'opération doit permettre de réaliser une économie globale de 40%" (exemple de Dial France, relaté dans 01 Références, Octobre 1992). "En interrogeant plusieurs entreprises, le cabinet KLC a constaté que leur motivation principale vis-à-vis du *facilities management* restait la réduction des coûts. Il l'évalue à plus de 25% dans un cas normal" (Le Monde Informatique, 12 Avril 1993).

Le critère des coûts est renforcé par le contexte de crise : "La crise est un vecteur de croissance pour le *Facilities Management*. Une société qui serre les boulons se laisse tenter par des économies attendues (promises ?) de l'ordre de 10 à 20% sur cinq ans" (Le Monde Informatique, 12 Avril 1993).

Pourquoi les fournisseurs seraient-ils à même de proposer des coûts inférieurs à ceux de l'informatique interne ? Deux explications sont généralement avancées.

Premièrement, la possibilité pour les professionnels de bénéficier d'économies d'échelle et de gamme. Lacity et Hirschheim (1993) résument cet argumentaire des partisans de l'externalisation en une seule formule (qu'ils remettent d'ailleurs en cause) : "*IS is a utility*". Les systèmes d'information seraient devenus une **ressource standardisée, homogène, dont la production peut être industrialisée** et les coûts amortis sur un volume de production important.

Le deuxième argument fait référence à la **spécificité technique** de la ressource. Outre les économies d'échelle, les professionnels du marché informatique mettent en avant les autres avantages classiques de la spécialisation industrielle : gains dus à l'expérience acquise dans la maîtrise de technologies complexes, l'organisation de sites d'exploitation, la gestion de projets et le contrôle des activités des informaticiens.

Mais au-delà de la promesse de baisse des coûts, nous pensons que l'idée de l'externalisation séduit parce qu'elle s'inscrit en résonance avec les thèmes actuels de l'évolution de l'informatique :

- tant que l'informatique bénéficiait d'une image scientifique forte et d'un statut de technologie de pointe, on pouvait admettre qu'elle soit gérée par une classe d'experts ; la direction informatique interne remplissait ce rôle. Dès lors que l'informatique est perçue comme une ressource banalisée, une production industrielle peut être envisagée, confiée à un secteur économique spécialisé,

- la volonté très forte des entreprises de contrôler les coûts de leur informatique et les carences traditionnelles de l'informatique interne en ce domaine ouvrent la voie aux sous-traitants,

- la prise de pouvoir par les utilisateurs peut conduire ceux-ci à s'adresser directement aux fournisseurs extérieurs,

- la rupture technologique et le passage aux architectures de type client-serveur conduisent certaines entreprises à envisager des solutions temporaires d'externalisation, soit pour libérer du temps aux équipes internes, soit au contraire pour bénéficier d'une expertise technique pendant le temps de mise en place de la nouvelle architecture,

- enfin le contexte de crise apparaît comme un catalyseur, tant pour les entreprises qui doivent renforcer l'accent sur le contrôle des coûts que pour les fournisseurs qui cherchent de nouveaux débouchés.

III. Quelles sont les limites de l'externalisation ?

De nombreuses descriptions de cas d'externalisation ont été publiées dans la presse professionnelle, mais il semble cependant difficile d'utiliser directement les informations ainsi obtenues. Lacity et Hirschheim, dont l'ouvrage "Information Systems Outsourcing : Myths, Metaphors and Realities" (1993) est sans doute la meilleure référence actuelle sur le sujet, ont cherché à rétablir la réalité sur les avantages et limites effectifs de l'externalisation.

Leur conclusion est que la vision donnée par la presse est exagérément optimiste, ce pour trois raisons: tout d'abord, les articles sont en général rédigés pendant la "lune de miel" qui suit immédiatement la signature du contrat. Ensuite, les économies qui sont annoncées sont les gains de coût *espérés* et non réalisés. Enfin, on parle très peu des échecs de l'externalisation. Le message généralement diffusé, qui est que l'externalisation va devenir la pratique dominante dans les années à venir, doit donc être sérieusement questionné. En particulier, concluent les auteurs, il est indispensable que des études de cas sérieuses soient menées, indépendamment des vendeurs de ces prestations.

Les deux chercheurs dénoncent l'effet de mode actuel ("*band-wagon effect*") qui caractérise la diffusion des pratiques d'externalisation au sein des différents secteurs de l'activité économique. Ils notent que la plupart des expériences positives relatées dans la presse correspondent aux premiers contrats d'externalisation et doutent qu'elles soient reproductibles pour les contrats suivants. Leur argument est que les fournisseurs consentent des efforts particuliers pour assurer la signature d'un premier contrat important avec une entreprise phare d'un nouveau secteur industriel. Ce contrat prend la forme d'un véritable partenariat avec l'entreprise considérée et les fournisseurs sont prêts à garantir -à perte- une qualité de service exceptionnelle, ce à des fins publicitaires. Ils se forgent ainsi rapidement une réputation de spécialiste du secteur (qu'ils ne seront effectivement au mieux qu'au bout de plusieurs années), démarchent ensuite les concurrents et s'appuient sur leur nouvelle réputation pour annoncer des économies d'échelle importantes et garantir leur efficacité. Cependant, passé cette première étape de séduction au sein du secteur, les contrats sont nettement moins favorables et il n'est pas possible aux nouveaux clients d'obtenir les mêmes résultats que ceux qui leur ont été présentés par les vendeurs et dont la presse a fait écho.

Concernant les coûts, les deux auteurs considèrent qu'un fournisseur extérieur n'est pas forcément plus efficient que le département interne et remettent en cause l'applicabilité du principe des économies d'échelle sur différents points, comme la taille du centre de calcul ou le coût des licences (compte tenu des nouveaux modes de facturation). Ils estiment donc que les

départements internes peuvent améliorer leurs performances et réaliser les mêmes économies qu'un fournisseur externe. Comme les fournisseurs extérieurs ne sont pas structurellement en position plus favorable que l'informatique interne, celle-ci peut mettre en oeuvre des actions pour améliorer la productivité et parvenir aux mêmes résultats.

Ils mettent en garde les entreprises contre les risques liés à l'externalisation et précisent que, si une entreprise décide d'externaliser tout ou partie de son système d'information, **le contrat est sa seule garantie**. En particulier, l'idée selon laquelle il est possible de travailler en partenariat avec un vendeur informatique est démentie car, contrairement au véritable partenariat dans lequel des entreprises s'associent pour attaquer une cible commune et partager des profits, il n'y a pas de communauté d'intérêt entre un "outsourcer" et son client. "*Un dollar dans la poche du fournisseur est un dollar pris dans la poche du client*" (p.259, traduit par nous). Il est donc nécessaire de fixer un maximum de détails **avant** le démarrage du contrat.

Lacity et Hirschheim réfutent la thèse selon laquelle les systèmes d'information formeraient un produit homogène et banalisé. La métaphore qui conduit à assimiler les systèmes d'information à un service banalisé comme l'eau, le gaz ou l'électricité ne peut se justifier (et encore) qu'en ce qui concerne les télécommunications ou le traitement de données. En particulier, le développement d'applications leur semble comporter trop de spécificités et nécessiter une communication trop étroite entre les informaticiens et le reste de l'entreprise pour qu'un tel raisonnement puisse s'appliquer.

Enfin, le "message clé" délivré à l'issue de leur ouvrage est que, s'il est envisageable d'externaliser certaines fonctions informatiques, il n'est en revanche "pas possible d'externaliser le management des systèmes d'information", au sens où l'entreprise doit conserver la maîtrise de son système, décider elle-même des objectifs qui lui sont assignés et ne pas se placer en situation de dépendance vis-à-vis de son fournisseur.

Conclusion

Nous avons cherché dans ce chapitre à comprendre le mouvement récent qui s'est déclaré en faveur de l'externalisation de l'activité informatique.

L'analyse de l'histoire de la gestion de la ressource informatique dans les entreprises nous a permis de montrer comment s'était constitué dans les années 70 un mode de gestion principalement interne sous la responsabilité des directions informatiques, favorisé par un ensemble de facteurs : image scientifique forte de l'informatique, dépendance des utilisateurs vis-à-vis de l'expertise des informaticiens, centralisation de la ressource, domination des marchés par les constructeurs, jeunesse de l'offre de services. Après la période d'adaptation et de diffusion de la micro-informatique qui a caractérisé les années 80, il semble que l'on arrive au moment de la **rupture** avec l'informatique traditionnelle :

- image de l'informatique maintenant banalisée,
- rupture technique avec la mise en place du modèle client-serveur,
- "désacralisation" du métier d'informaticien,
- volonté de confier la responsabilité de la ressource informatique aux utilisateurs,
- recherche de l'efficacité (atteinte des objectifs au moindre coût),
- crise dans la croissance avec la limitation sévère des budgets informatiques.

Le discours des partisans de l'externalisation est en résonance avec les différents aspects de cette crise à laquelle il propose un nouveau type de solutions :

- pour une ressource dont la spécificité technique reste forte mais qui est maintenant banalisée dans les esprits, il propose un schéma classique de spécialisation industrielle : à chacun son métier, la spécialisation permettant à la fois des économies d'échelle et la constitution d'un savoir-faire chez les professionnels,

- au problème de contrôle des coûts internes de l'informatique, il propose une solution radicale d'externalisation et un arbitrage par le jeu de la concurrence sur le marché,

- aux conflits éventuels de pouvoir entre informaticiens et utilisateurs, il propose l'arbitrage par une tierce partie (le fournisseur extérieur),

- aux services informatiques internes surchargés par le poids des maintenances et inquiets devant l'évolution technologique vers une nouvelle architecture, il propose des solutions de prise en charge ou de relais temporaire,

- aux directions générales à la recherche de moyens de limitation des coûts, il propose des réductions spectaculaires des budgets.

De nombreux aspects de ce discours peuvent certes être remis en cause ; les risques n'y sont pas évoqués et les avantages promis ne sont pas toujours tenus. Il reste que l'externalisation fait maintenant partie des solutions reconnues de gestion de l'informatique et que les experts prévoient son développement dans les années à venir. Sous quelle forme ? Sans doute pas sous celle de contrats globaux fortement médiatisés où une entreprise fait appel à un fournisseur unique pour l'ensemble des activités informatiques. Plus vraisemblablement, les entreprises seront amenées à moduler l'externalisation en fonction des objectifs et du type d'activité informatique concerné. Pour notre part, nous avons choisi de nous intéresser spécifiquement à l'activité de développement d'applications, dont l'externalisation, au dire des chercheurs, présente des spécificités que nous allons étudier dans le chapitre 2.

Chapitre 2 :
l'externalisation du développement d'applications

Introduction

Après avoir analysé le contexte général de crise que vit à présent l'informatique et étudié la logique du mouvement récent en faveur de l'externalisation de cette fonction, nous voulons ici nous focaliser sur notre objet d'étude, l'activité de développement d'applications.

Pourquoi le choix de cet objet d'étude ?

La première constatation qui frappe le chercheur s'intéressant à cette question est la quasi absence du thème de l'externalisation dans les ouvrages, pourtant très nombreux, traitant du développement d'applications. La question de la ressource de développement est un thème classique de la littérature en systèmes d'information, mais qui n'a longtemps été traité que dans une perspective **interne**, même si, comme nous l'avons vu, on pouvait constater dans la pratique dès les années 70 que les entreprises faisaient parfois appel à la sous-traitance auprès des SSII. A l'heure actuelle encore, les manuels de la discipline détaillent longuement les différents aspects du développement d'applications en interne mais omettent souvent d'aborder le sujet des ressources extérieures en matière de logiciel¹.

La question a été posée dans un premier temps dans le cadre restreint de l'acquisition de progiciels applicatifs. Le choix entre développement spécifique ou progiciel (faire ou acheter), a été largement exploré dans la littérature. De nombreux chercheurs se sont attachés à décrire les modalités de cet arbitrage ; nous exposerons les principaux résultats obtenus dans la première section de ce chapitre.

On a pu penser que les développements spécifiques disparaîtraient au profit des progiciels. Cependant, ceux-ci ne forment pas une réponse complète aux besoins de développement d'applications : leur adaptation aux besoins de l'entreprise et leur intégration au système

¹. On peut noter à titre d'exemple que l'excellent ouvrage de Davis et al. "Systèmes d'information pour le management" (1986), qui constitue par ailleurs une référence pour l'enseignement de cette matière, n'aborde tout simplement *pas* le thème du recours aux fournisseurs extérieurs, que ce soit sous l'angle de la sous-traitance ou de l'acquisition de progiciels.

d'information génère en elle-même des besoins de développements ; de plus, ils ne peuvent pas prendre en compte tous les développements applicatifs.

Il reste donc une partie (parfois majoritaire) de l'activité de développement d'applications non satisfaite par des produits standards. Externaliser le développement d'applications pose alors la **question du recours à un fournisseur extérieur pour des travaux spécifiques** : développement d'une nouvelle application, maintenance, interfaces et développements complémentaires autour d'un progiciel. Cette question a été très peu étudiée et pose des problèmes particuliers, en partie différents de ceux que nous avons constatés en analysant la question de l'externalisation globale de l'informatique : **c'est donc le domaine que nous avons choisi d'explorer**, dont nous définissons les contours dans la deuxième section de ce chapitre.

Section 1 - L'acquisition des progiciels : un arbitrage davantage centré sur la nature du produit que sur l'origine de la ressource

I. A l'origine de la diffusion des progiciels : le "goulot d'étranglement du logiciel"

Le "goulot d'étranglement du logiciel" (Gremillion et Pyburn, 1983), la "crise du logiciel" (McFadden et Discenza, 1987) sont des expressions maintenant familières qui sont apparues quand les progrès énormes enregistrés dans le domaine du matériel informatique ont fait passer le logiciel au premier rang des préoccupations des professionnels. A partir du moment où la "loi de Grosch" (la puissance d'un ordinateur croît avec le carré de sa taille et son prix croît avec sa taille) est abolie du fait de la miniaturisation des circuits et de l'apparition de la micro-informatique, s'enclenche un mouvement lent mais irrésistible qui va provoquer la chute des géants du matériel informatique, la baisse des coûts et l'ascension de l'industrie du logiciel.

Comme le signale un rapport de l'OCDE en 1985, "Les progrès spectaculaires réalisés dans les constituants matériels des technologies de l'information n'ont une utilité économique et sociale que dans la mesure où ils sont complétés par des logiciels de niveau adéquat, susceptibles de les rendre opérationnels et de les rendre accessibles à la majorité des acteurs économiques et sociaux. C'est la constatation qu'il peut y avoir de sérieux problèmes techniques et économiques qui pèsent sur le développement et la diffusion de cette partie logicielle qui a donné naissance aux premières craintes sur un éventuel goulot d'étranglement du logiciel"².

Le problème semble au premier abord quantitatif (c'est l'époque où l'on cherche à former de toute urgence des programmeurs) ; en fait il est également (et peut-être surtout) qualitatif. La

². Les logiciels : l'émergence d'une industrie, rapport de l'OCDE, 1985.

productivité est très faible, les outils d'aide à la programmation sont limités. Les constats que l'on peut faire dans les entreprises vers la fin de la décennie 80 sont inquiétants :

- le volume de projets en attente (*backlog*) est particulièrement élevé. McFadden et Discenza (1987) parlent d'un délai d'attente de 3 à 4 ans pour les développements internes, Chokron et Reix (1987) de retards de 2 à 3 ans pour les services études, Venner (1988) de demandes en attente représentant l'équivalent de 3 à 4 années des ressources internes.

- des modifications anodines aux yeux des utilisateurs prennent des mois avant d'être effectuées. La charge de maintenance devient insupportable et absorbe une grande partie de l'énergie de développement.

- le cycle de production des logiciels est particulièrement lent. Il n'est pas rare que de gros projets s'échelonnent sur plusieurs années. Rands (1991a) souligne le contraste entre des besoins qui évoluent rapidement sous la pression d'un environnement dynamique et de très longs délais de production des applications. Le cycle de production des logiciels internes devient plus long que le cycle d'activité de l'entreprise, situation qui conduit à une impasse et va pousser les organisations à examiner toutes les solutions alternatives.

- le processus de développement en usage (cycle de la cascade³) ne parvient pas à saisir l'essence des besoins des utilisateurs et se montre très peu flexible. Du coup les systèmes développés "tendent à refléter ce dont les utilisateurs *pensaient* avoir besoin il y a *un an* plutôt que ce dont ils *ont réellement* besoin *aujourd'hui*" (McFadden et Discenza, 1987). Parmi les utilisateurs et dans les services informatiques circulent des dessins humoristiques du type de celui présenté dans le très sérieux rapport de l'OCDE et que nous ne résistons pas à la tentation de reproduire⁴ à notre tour.

³. On trouvera dans le chapitre 3 une description détaillée du processus de développement et la présentation du cycle de la cascade.

⁴. Ne serait-ce que parce que j'ai vu ce dessin affiché sur la porte d'un client pendant plusieurs années consécutives, où il suscitait de nombreux commentaires parmi les utilisateurs (généralement approubateurs).

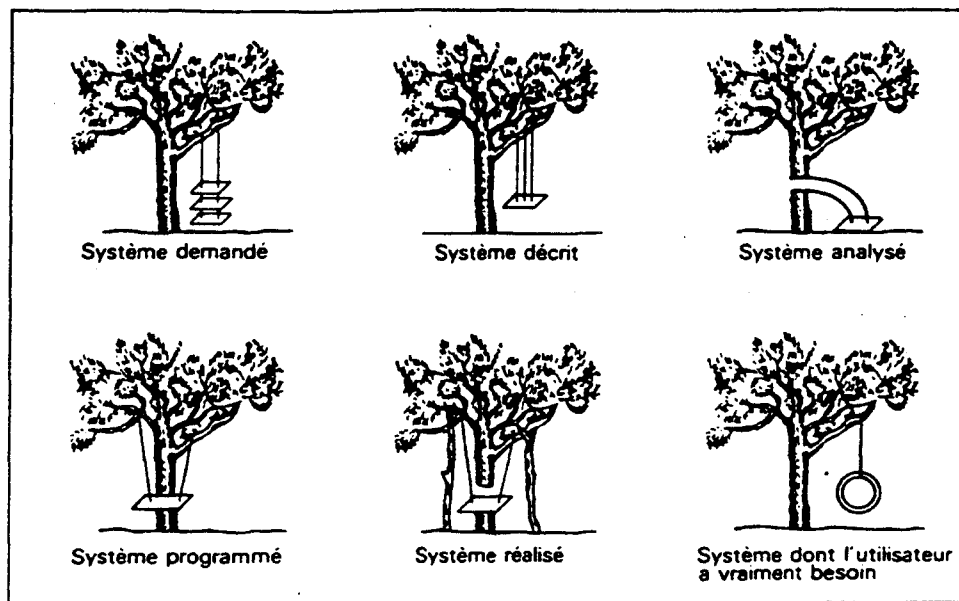


Figure 2.1 Représentation courante du problème de conception des logiciels : métamorphoses d'un système au cours de son élaboration (source : OCDE, 1985)

- le taux de rotation du personnel informatique est particulièrement élevé. Il semble très difficile de recruter, de motiver et de fidéliser les informaticiens au sein des services internes. Ce facteur est d'autant plus inquiétant que des études ont montré que la compétence des programmeurs est le principal facteur le plus jouant sur la productivité dans le domaine du logiciel (Boehm a ainsi identifié des écarts de productivité allant de 1 à 4).

- le coût des développements ne cesse de croître. D'une part les demandes en logiciel sont de plus en plus sophistiquées et d'autre part les développeurs continuent à utiliser des langages assez anciens comme COBOL (50% de l'utilisation totale des langages aux Etats-Unis en 1985 selon IBM), FORTRAN ou BASIC. La productivité est faible.

- l'ensemble de ces dysfonctionnements tend à ruiner les efforts pourtant répétés de planification informatique. On est à l'époque des schémas directeurs à horizon de 3 à 5 ans qui seront délaissés par la suite au profit d'une vision plus adaptative et réactive⁵.

Face à cette situation, plusieurs solutions sont envisagées. Certains utilisateurs commencent à réaliser de petits développements (Gremillion et Pyburn, 1983). Les infocentres vont ouvrir la voie et la tendance va se renforcer avec l'apparition de la micro-informatique. L'informatique devient plus accessible et les utilisateurs commencent à prendre en charge certaines décisions auparavant du domaine réservé des informaticiens (achat de matériel et de logiciel micro, accords avec des fournisseurs). Le service informatique n'a plus l'exclusivité sur les achats. Ce phénomène prend de l'ampleur mais reste difficile à évaluer (Rands, 1990).

Du côté de l'informatique interne "classique", Hayward (1987) identifie trois approches possibles, qui, combinées, pourraient constituer une réponse au problème de productivité des développements. Les deux premières visent à améliorer la productivité interne, la dernière **introduit le recours aux sources externes** :

- améliorer la productivité en utilisant des outils d'aide au développement,
- changer de génération de langage informatique en utilisant des langages de "quatrième génération", plus conviviaux⁶,
- acheter les progiciels applicatifs.

C'est vers cette dernière solution que vont se tourner le plus rapidement les entreprises. Ce sera l'occasion de poser vraiment pour la première fois la question de l'arbitrage entre ressources internes et externes.

⁵. Les systèmes d'information sont loin d'être le seul domaine à être concerné par cette évolution. Il s'agit en fait d'une tendance générale du management des toutes dernières années qui prône la flexibilité, l'adaptation au changement et la réactivité, considérés comme plus efficaces dans un environnement économique incertain et mouvant. Même si ce n'est pas incompatible avec une planification, une "gestion de l'instabilité" s'impose.

⁶. Dans la pratique, les langages de 4ème génération (L4G) sur gros système (Focus, Nomad ..) ne connaîtront pas le succès espéré : ils arrivent trop tôt en termes de capacité du matériel informatique (ce sont d'énormes consommateurs de ressources). La nouvelle génération de langages micro-informatiques, comme les tableurs, connaîtra, elle, un très beau succès auprès des utilisateurs pour de petits développements simples, alors que les programmeurs seront tentés par les outils logiciels micro-informatiques, notamment graphiques.

II. Avantages et inconvénients des progiciels

La littérature sur le choix "progiciel / développement interne" envisage généralement ce choix dans une perspective classique d'achat industriel, en prenant en compte les critères de capacité, de coût de production et de délai. Cette approche permet de dégager un ensemble d'arguments **en faveur** des progiciels, récapitulés dans le tableau 2.2.

Le premier avantage évident est la réponse aux problèmes de capacité. L'informatique interne est surchargée et l'on pense au départ que quelques progiciels -limités à des fonctions très standardisées comme la comptabilité ou la paie- vont permettre de dégager du temps pour la masse prépondérante d'applications spécifiques. Cependant, assez rapidement, les avantages en termes de coût et de délais conduisent certains auteurs à recommander aux organisations d'explorer systématiquement une solution progiciel (Sibley, 1985), le développement spécifique interne ne devant être envisagé qu'en cas d'impossibilité de cette première solution. Cette recommandation prend un caractère d'obligation pour les petites entreprises, pour lesquelles le progiciel apparaît comme la seule solution accessible (Martin et McClure, 1983, Rands, 1990).

Les avantages de l'acquisition d'un progiciel	
Réponse aux problèmes de capacité (<i>backlog</i>)	Gremillion (1982), Gremillion et Pyburn (1983), Martin et McClure, (1983), Bryce et Bryce (1987), Venner (1988), Lucas et al. (1988), Gershkoff (1990)
Baisse des coûts	
• coût d' <i>acquisition</i> faible (économies d'échelle)	Gremillion (1982), Martin et McClure (1983), Sibley (1985), Gershkoff (1990), Blanc (1992)
• moins de coûts <i>cachés</i>	Sibley (1985)
• moins de <i>dépassements</i> imprévus	Sibley (1985), Gershkoff (1990), Blanc (1992), Sherer (1993)
• coûts de <i>maintenance</i> réduits	uniquement si le progiciel est flexible : Martin et McClure, (1983), Blanc (1992)
• baisse des <i>coûts de personnel</i>	Blanc (1992)
Réduction des délais	Gremillion (1982), Martin et McClure (1983), Sibley (1985), Bryce et Bryce (1987), Blanc (1992), Sherer (1993)

Tableau 2.1. L'approche industrielle du choix entre progiciel et développement interne

Au départ envisagés comme un palliatif pour répondre à une situation de sous-capacité, les progiciels s'imposent donc comme une solution attractive à envisager en priorité. Pour beaucoup d'entreprises, la règle est maintenant de privilégier les "solutions progiciel", **indépendamment du niveau de ressource interne disponible**. Du même coup, la situation de sous-capacité que vivaient les équipes internes de programmation pourrait se transformer assez rapidement en situation de surcapacité. Bien que l'on ne dispose pas d'études chiffrées directes sur une éventuelle sur-capacité de développement, le marché de l'emploi des programmeurs nous fournit un indicateur : le rapport offre/demande semble s'être inversé dans les toutes dernières années et les tendances salariales sont globalement à la baisse. Contrairement aux attentes exprimées dans les années 80, le chômage a fait son apparition dans cette catégorie de personnels auparavant épargnés.

Cependant, même si la "solution progiciel" est de plus en plus sérieusement envisagée, certains freins subsistent à l'utilisation systématique de ces produits.

Premier révélateur de ces freins, nous avons noté avec intérêt l'absence du thème de la **qualité** lors notre étude des avantages reconnus aux produits standard. La qualité n'est en effet abordée que sous l'angle de la critique (lacunes des produits, mauvaise adéquation aux besoins, retard technologique⁷) ou du risque. Ce point de vue nous semble assez sévère compte tenu des progrès récents des produits. Quoi qu'il en soit, il faut noter que les progiciels sont l'objet d'un *a priori* **négatif** quant à la qualité, qu'il s'agisse des programmes, de la documentation (Martin et McClure, 1983), de la stabilité et de la fiabilité des fournisseurs (Martin et McClure, 1983, Gershkoff, 1990) ou des services connexes proposés par ceux-ci.

De plus, les progiciels sont **une réponse standardisée** au problème de traitement de l'information et, en tant que tels, ne peuvent être adaptés que si la **spécificité du problème et de son traitement** sont limitées.

Gremillion et Pyburn (1983) proposent ainsi de restreindre le choix des progiciels aux projets acceptant une solution standardisée. Si la spécificité de la solution à mettre en oeuvre est trop élevée, d'une part on peut se trouver en situation de "non-choix" (progiciel inexistant) et d'autre part on peut choisir de rejeter l'utilisation d'un progiciel parce que les **coûts d'adaptation** interne seraient prohibitifs. Les auteurs suggèrent alors l'emploi d'autres moyens (développement traditionnel, développement avec prototypage, développement par l'utilisateur), mieux adaptés.

De nombreux auteurs soulignent les **difficultés lors de la mise en oeuvre** des progiciels. Sherer (1993) indique ainsi que ce type de solution réduit les risques liés à la phase de

⁷. Le parc actuel de progiciels comporte beaucoup de produits développés en Cobol. Même si les SSII suivent les avancées technologiques, on peut leur reprocher de n'investir en matière de progiciel que sur des technologies maîtrisées par leurs clients potentiels : "Dans l'assurance, les fournisseurs attendent que les utilisateurs maîtrisent les bases de données relationnelles comme DB2 pour développer leur offre. En faisant du spécifique, on est plus facilement à la pointe de la technologie" (le responsable de l'organisation et de l'informatique d'une filiale assurance du Crédit Agricole, cité par Blanc, 1992).

développement (faibles risques de gestion de projet, risque politique plus facile à gérer, peu d'aléas techniques), mais *augmente* ceux de la phase d'utilisation : risque d'inadéquation fonctionnelle et de mauvaise performance, risques liés aux changements futurs (maintenance, adaptation, portabilité). Les différents auteurs identifient deux origines principales du risque en phase d'utilisation.

a) Le premier facteur de risque est **l'insuffisance des procédures de sélection des produits**.

Lucas et al. (1988) ont cherché à déterminer les variables qui jouent sur le succès de l'installation d'un progiciel et ont testé empiriquement leur modèle sur un échantillon de 18 entreprises. Les résultats obtenus montrent que l'écart entre les besoins des utilisateurs et les fonctionnalités du produit joue un rôle important. Les auteurs recommandent donc de mener une véritable analyse conceptuelle des besoins avant de s'engager dans le choix d'un progiciel.

En effet, les procédures actuelles de sélection des produits ne permettent généralement pas d'évaluer les performances du progiciel *dans son environnement futur d'utilisation*. On a pu penser qu'il serait relativement simple de choisir un produit existant et que l'on éviterait ainsi les problèmes de conception illustrés par le dessin satirique repris par l'OCDE. Les faits opposent un démenti répété à cette croyance. Visualiser un produit, assister à une démonstration ou même rencontrer un autre utilisateur ne suffit pas pour déterminer si le progiciel satisfait aux besoins de l'utilisateur ni pour évaluer les changements d'organisation et de procédures nécessaires. La plupart des auteurs mettent donc en garde les utilisateurs contre des choix précipités (Martin et McClure, 1983), accordant trop d'importance au critère du prix du produit (Lucas et al., 1988), effectués de façon *ad hoc* sans vision globale des besoins à plus long terme ni perspective d'intégration au système d'information (Sibley, 1985), ou encore menés directement par les utilisateurs sans concertation avec les informaticiens internes (Bryce et Bryce, 1987). Ces deux auteurs insistent sur la nécessité de mener correctement l'étape d'évaluation et soulignent que "l'étude de faisabilité doit être exécutée de la même manière, qu'il s'agisse d'un développement interne ou de l'acquisition d'un progiciel". Gershkoff (1990) compare l'évaluation d'un progiciel à la phase finale de test d'un développement sur mesure

avant sa mise en exploitation et suggère d'utiliser les mêmes procédures. Par ailleurs, se mettent en place des méthodes rationnelles de choix des progiciels (Martin McClure, 1983, Sibley, 1985, Bryce et Bryce, 1987, Anderson, 1990, Le Blanc et Jelassi, 1991).

b) Le second facteur de risque est le **changement organisationnel** induit par l'introduction d'un progiciel.

Gremillion (1982) souligne dès l'origine l'importance de ce facteur et la rupture avec les habitudes de mise en oeuvre d'un logiciel spécifique. Dans le cas d'un développement "sur mesure", le parti pris est que le produit doit s'adapter aux besoins de l'utilisateur. Cette situation est renversée en partie dans le cas d'un progiciel. Dans la mesure où certaines fonctionnalités sont fixées, on va demander à l'utilisateur de *changer son comportement pour s'adapter au produit*.

Cet argument, qui s'exprime en défaveur des progiciels, a pu être retourné pour être considéré au contraire comme un atout potentiel de ces produits. On peut ainsi estimer que l'utilisation d'un progiciel évite une surenchère de la demande de fonctionnalités non indispensables de la part des utilisateurs (Blanc, 1992). Par ailleurs, l'entreprise peut dans certains cas bénéficier de l'expertise associée au produit : le changement conduira à améliorer les procédures. Hansen (1986) évoque le cas des progiciels verticaux (spécialisés par secteur d'activité comme la banque ou l'assurance) qui, au contraire des progiciels horizontaux (spécialisés par fonction - comptabilité, paie..) s'accompagne d'une expertise du vendeur sur le métier de l'acheteur. Rands (1990) a estimé la part des progiciels verticaux sur le marché anglais à 56% du nombre total de produits. Cependant, il est très difficile de tirer des conclusions d'un chiffre aussi global, qui cache des diversités importantes selon les secteurs et qui mêle différentes catégories de progiciels pour différentes tailles d'organisations. Rands estime cependant que la spécialisation verticale est l'une des tendances fortes du marché du logiciel.

Enfin, dernier frein à l'usage du progiciel, Martin et McClure (1983) évoquent le "syndrome du NIH" (*not invented here*), Bryce et Bryce (1987) le syndrome de "*reinvent the*

wheel" qui conduisent à rejeter un progiciel ou à ne pas envisager son emploi dans des circonstances où il serait pourtant économiquement justifié.

III. Le choix entre progiciel et développement spécifique interne : le modèle de Buchowicz

La plupart des auteurs dont nous avons évoqué les travaux mettent donc en avant les critères de **délai**, de **coût** et de **ressources disponibles** pour justifier l'acquisition de progiciel. A l'inverse, l'**inadéquation des logiciels standard** et l'obligation de **modifier les procédures et l'organisation interne** pour les utiliser constituent des freins à l'acquisition de ces produits. Cependant, si l'on excepte les travaux de Lucas et al. (1988) sur l'installation de progiciels, ces recherches manquent de validation empirique solide. Buchowicz (1991) a cherché à combler ce manque en menant l'étude qualitative du processus de décision dans six organisations. La décision étudiée est le choix entre l'acquisition ou le développement en interne d'un logiciel de gestion de production. L'auteur déduit de ses observations le modèle de choix présenté dans la figure 2.2.

Le processus de décision comporte selon Buchowicz trois étapes:

- (1) évaluation de l'adéquation stratégique du choix (*strategic fit evaluation*)
- (2) première orientation du choix à partir de quatre critères (*initial categorization*)
- (3) décision (*selection*). Cette dernière étape comporte elle-même deux phases : dans un premier temps, la décision finale entre achat ou développement interne est affinée à partir de trois critères supplémentaires. Dans un deuxième temps, l'auteur explore les modalités du choix d'un fournisseur particulier dans le cas d'une décision d'achat ("*Buy/buy*" choice, ou encore processus plus ou moins formalisé d'arbitrage entre des fournisseurs différents, que nous ne développons pas ici).

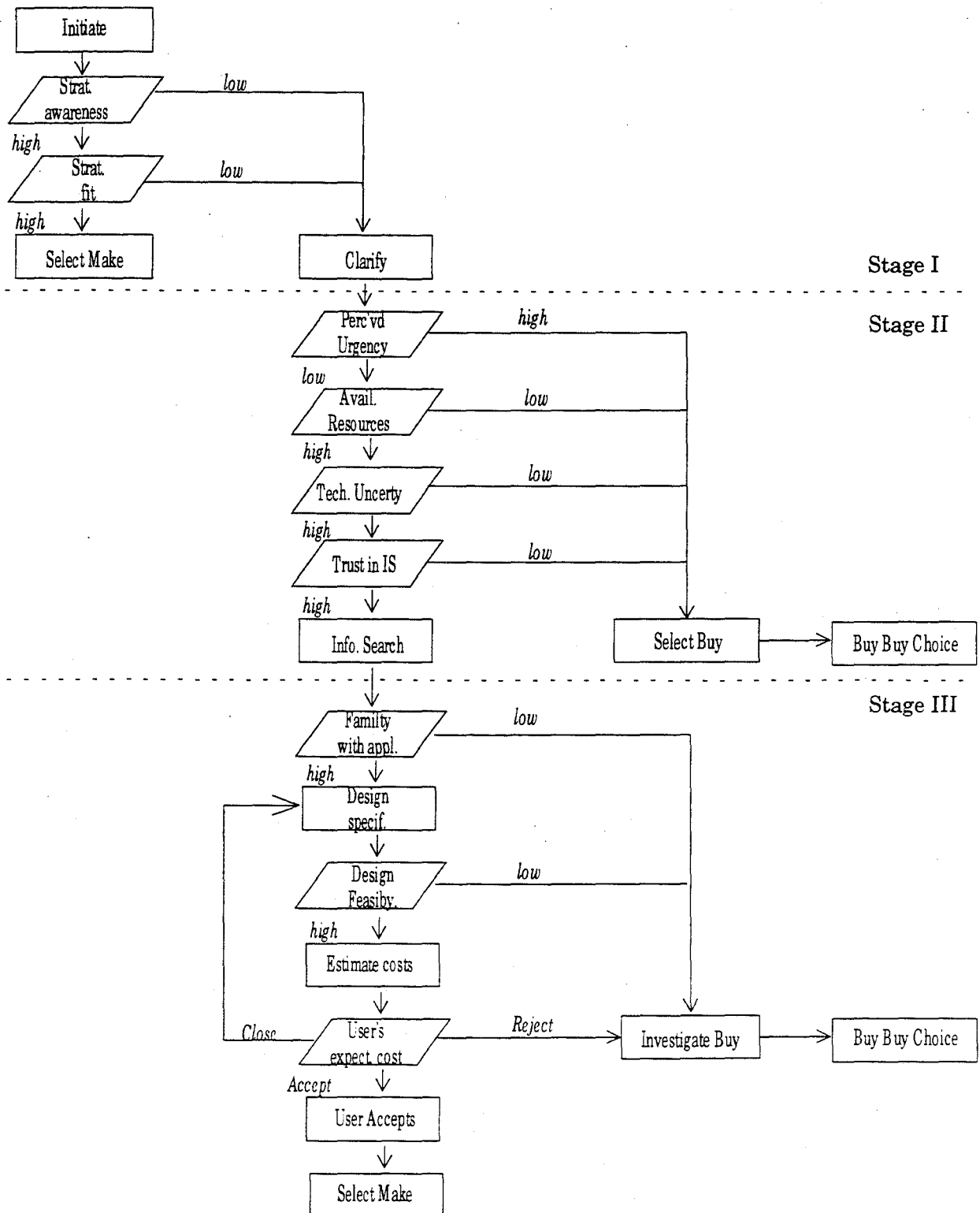


Figure 2.2. L'arbitrage entre développement interne et acquisition de progiciel :
le modèle de Buchowicz (1991)

L'apport majeur du modèle de Buchowicz est de présenter le choix comme une série de décisions séquentielles et de permettre de **hiérarchiser les critères** jouant sur la décision. Cette hiérarchie apparaît au travers des dix règles de décisions concernant le choix final :

Règle 1 : Le développement en interne sera retenu si le choix est effectué par une personne connaissant bien les options stratégiques de l'entreprise et si le développement interne offre une opportunité de création d'un avantage concurrentiel (première étape du choix).

Dans les autres cas, les règles 2 à 5 s'appliquent (deuxième étape du choix).

Règle 2 : L'achat sera retenu si la décision est perçue comme urgente.

Règle 3 : L'achat sera retenu s'il y a carence quantitative ou qualitative des ressources internes (matériel, logiciels de développement, équipes).

Règle 4 : L'achat sera retenu si la technologie utilisée par les progiciels est stabilisée et si les approches standard s'imposent.

Règle 5 : L'achat sera retenu si la confiance des utilisateurs dans la capacité de l'équipe informatique interne à réussir le contrat est faible.

A noter qu'*une seule* de ces conditions vérifiée conduit selon l'auteur à choisir l'achat. Dans le cas contraire, le développement interne doit encore franchir un ensemble de barrières pour être retenu (étape 3, règles 6 à 10) :

Règle 6 : L'achat sera retenu si le service informatique interne refuse de développer parce qu'il ne connaît pas bien le domaine applicatif et estime donc que les risques d'échecs sont trop importants.

Règle 7 : L'achat sera étudié si l'étape de conception générale ne débouche pas sur un projet réalisable.

Règle 8 : L'achat sera étudié si les estimations de coût fournies par l'informatique interne dépassent largement le budget maximum alloué par l'utilisateur.

Règle 9 : Si les estimations de coût dépassent légèrement le budget maximum alloué par l'utilisateur, l'informatique interne cherchera à réduire les coûts en adaptant la conception.

Règle 10 : Le développement en interne sera retenu si les nouvelles estimations de coût fournies par l'informatique interne satisfont l'utilisateur.

Les travaux de Buchowicz comportent un ensemble de résultats très intéressants :

1. Le critère de l'adéquation est identifié comme le critère majeur de choix dans une décision de type *make/buy*
2. Si le projet n'a pas un fort intérêt stratégique, le choix est fortement biaisé en faveur d'une solution externe. Il ne faut pas moins que la conjonction de sept critères tous favorables pour que le développement interne soit retenu, alors que la défaillance d'un seul critère incite à sélectionner l'acquisition.

Le biais à l'achat si le projet n'est pas jugé stratégique		
Choix	Critère	Condition sur le critère
Dévelop. Interne si..	Importance stratégique perçue.....	forte
	ou critères 2, 3, 4, 5, 6, 7 et 8.....	tous favorables
Achat si..	1. Délai, urgence.....	forte
	ou 2. Disponibilité de la ressource interne.....	mauvaise
	ou 3. Incertitude technologique.....	faible
	ou 4. Confiance en l'informatique interne.....	faible
	ou 5. Connaissance du domaine applicatif.....	faible
	ou 6. Faisabilité après la conception générale.....	faible
	ou 7. Coût du projet en interne.....	trop élevé

3. En étudiant les conditions évoquées par Buchowicz, on peut identifier plusieurs niveaux d'analyse de la décision, correspondant à des groupes de variables :

- la plupart des variables sont à rattacher au **projet** lien avec la stratégie (*strategic fit*), délai, incertitude technologique, domaine applicatif déjà connu ou non, faisabilité, coût.

On peut rapprocher certaines de ces variables de celles utilisées par Gremillion et Pyburn (1983) : la connaissance *a priori* du domaine applicatif détermine largement le degré de structure du produit, l'impact est vu ici sous son angle stratégique. En revanche, la spécificité du problème et des solutions adoptées n'est pas reprise directement dans le modèle de Buchowicz.

- une variable traduit l'état de la **ressource interne** (disponibilité)

en fait, on pourrait scinder cette variable en deux sous-éléments qui sont la quantité de ressource disponible et la qualité de cette ressource, adaptée ou non.

- deux variables traduisent la **culture interne** de l'organisation

La première est l'existence ou non d'une culture **stratégique** largement diffusée dans l'organisation (*strategic awareness*) : certaines firmes ont une approche stratégique de la décision (étape 1 du choix), alors que d'autres adoptent une approche **tactique** du choix, omettent la première étape et ne prennent en compte que des critères opérationnels. Pour Buchowicz, les firmes adoptant une approche stratégique privilégient l'efficacité (atteinte des objectifs à long terme), celles ayant une approche tactique privilégient l'efficacité (optimisation des coûts à plus court terme). Le modèle développé par l'auteur indique que pour les firmes à culture tactique la solution d'acquisition sera privilégiée, alors que pour les firmes à culture stratégique le développement interne pourra être retenu indépendamment de considérations opérationnelles.

La deuxième variable culturelle est le climat de confiance qui s'est établi ou non entre les utilisateurs et l'informatique interne.

Les résultats de Buchowicz n'apportent qu'une validation modérée de la théorie classique d'achat industriel. Les questions de coût de production, de délai ou d'optimisation du niveau de la ressource, si elles sont évoquées, ne sont pas considérées comme majeures. L'auteur s'inscrit plutôt dans une approche **stratégique de la gestion de la ressource**, qui consiste à intégrer en interne les activités qui sont perçues comme étant au coeur du métier de l'organisation et à ne pas hésiter à externaliser les activités secondaires si cela présente un avantage en termes de coût, de qualité, de délai ou de choix d'investissements.

En conclusion sur les travaux de Buchowicz nous soulignerons que cette recherche est à bien des égards pour nous exemplaire. Tout d'abord, en se focalisant sur l'analyse de décisions précises l'auteur évite de se limiter à un discours généraliste sur les coûts, la ressource et les

délais et parvient à mettre en évidence le rôle d'autres facteurs. Ensuite, le choix d'un petit nombre d'organisations lui permet d'analyser des facteurs difficilement quantifiables comme la diffusion d'une culture stratégique et le climat de confiance entre utilisateurs et informaticiens. Enfin, le modèle permet de dégager autour du projet, qui est le niveau d'analyse central, l'impact de facteurs "d'environnement" : la disponibilité des ressources internes et l'environnement organisationnel (qualité des relations entre informaticiens et utilisateurs, culture stratégique interne).

Cependant, comme l'auteur le signale lui-même, il s'agit d'une recherche exploratoire et le faible nombre de cas étudiés ne permet pas de prouver la validité externe du modèle.

Une autre limite de la recherche doit être soulignée. Le choix d'externalisation étudié (*make versus buy*) est présenté sous une forme binaire : le "**tout faire**" sous la forme du développement spécifique interne et le "**tout acheter**" sous la forme du progiciel. C'est d'ailleurs la présentation commune adoptée par l'ensemble des travaux portant sur le choix d'un progiciel.

Nous estimons pour notre part que cette présentation donne une vision simplificatrice de la réalité : quand une organisation achète un progiciel, il reste un ensemble de travaux (interfaces, paramétrage, éventuels développements complémentaires, installation) pour lesquels se posent également le problème du choix entre ressource interne et externe. Il est difficile d'évaluer la part représentée par l'achat du produit dans le coût global d'un projet à base de progiciel. Un associé d'Arthur Andersen France nous a donné une estimation de 1/5 des coûts totaux. Parmi les 4/5 restants, une partie peut être sous-traitée mais il peut également y avoir beaucoup de ressources internes engagées, si bien que la qualification "tout acheter" paraît abusive.

C'est pourquoi ne prendre en compte que la partie du projet représentée par l'acquisition du progiciel est insuffisant pour qualifier la décision de solution d'achat pure, sauf dans le cas de produits très simples et peu intégrés au système d'information. Il est donc nécessaire d'affiner l'analyse du choix et on ne peut pas se limiter à deux modalités pour définir la décision.

Enfin, l'ensemble de ces recherches sur les progiciels **lie l'arbitrage interne / externe à l'arbitrage produit spécifique / produit standard**. Les deux problématiques sont implicitement confondues. Il est cependant envisageable d'acquérir un logiciel spécifique en confiant sa réalisation à l'extérieur, ou d'envisager des formes intermédiaires (progiciel et adaptations complémentaires) mêlant ressource interne et ressource externe. Ces choix ont été explorés par Rands, chercheur dont nous allons maintenant analyser les travaux.

IV. Les travaux de Rands sur les choix intermédiaires

Rands a mené un ensemble de travaux intéressants sur l'arbitrage interne/externe dans le cas des logiciels (1990, 1991a et surtout 1991b). Nous avons déjà évoqué ses remarques sur l'évolution du secteur vers des marchés verticaux (1990) ainsi que son analyse de l'influence sur la décision d'achat d'un progiciel de temps de production trop élevés des logiciels internes (1991a). Nous allons maintenant étudier un autre de ses recherches, à savoir sa réflexion originale sur les formes intermédiaires entre ressource externe et interne.

Rands nuance le choix "acheter / faire" défini par la réflexion sur les progiciels en introduisant des formes intermédiaires dérivées de l'une et l'autre de ces solutions. Plutôt qu'un choix binaire, on se retrouve alors en face d'un continuum de choix tendant à plus ou moins d'externalisation (fig. 2.3)

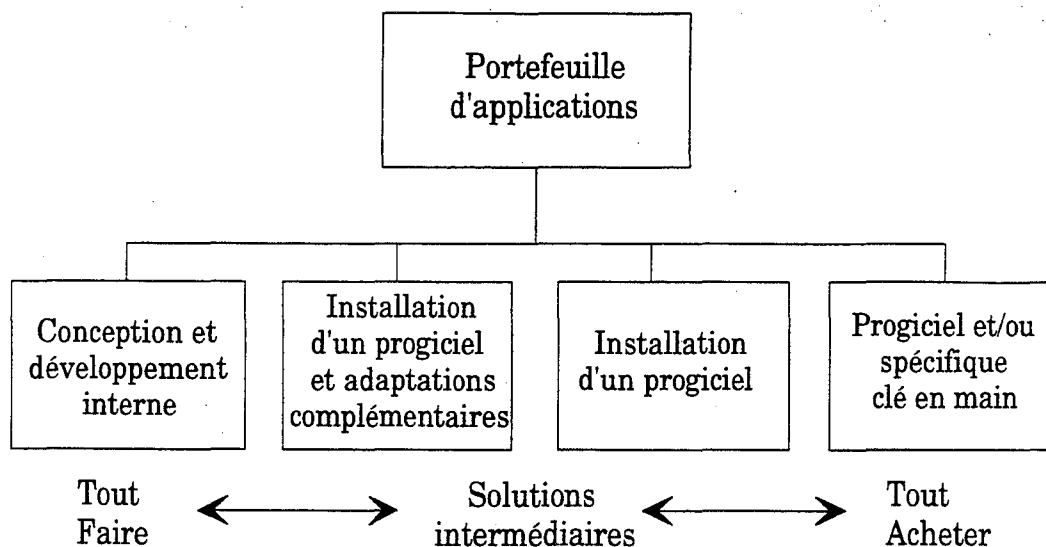


Figure 2.3. Les choix intermédiaires entre tout faire et tout acheter (source : Rands, 1991b)

- Les formes intermédiaires dérivées du progiciel prennent en compte l'installation, les adaptations et compléments autour de progiciels.

- Les formes intermédiaires dérivées du développement spécifique interne prennent en compte la possibilité pour la firme de sous-traiter tout ou partie des composantes d'un projet. Rands signale ici les deux formes extrêmes (développement entièrement interne et sous-traitance d'un spécifique clé en main).

Contrairement aux travaux précédents sur les progiciels, on aboutit ainsi à une dissociation nette entre les deux problématiques "standard/spécifique" et "interne/externe". Certaines formes de développements spécifiques sont complètement internalisées et d'autres totalement externalisées. Certaines formes de progiciels peuvent faire assez largement recours à la ressource interne.

Section 2 - Notre champ d'étude : l'arbitrage entre ressource interne et externe pour des travaux spécifiques

I. La définition du champ

Nous avons choisi d'étudier les choix d'externalisation pour des travaux de développement d'applications spécifiques.

Deux raisons essentielles ont motivé ce choix :

- la première raison est l'existence d'une **carence des recherches** en ce domaine. Alors que la question de l'acquisition des progiciels a donné lieu à de nombreuses publications et à l'élaboration de modèles de choix satisfaisants, l'étude de l'externalisation des travaux spécifiques en est encore à ses premiers pas.

- la seconde raison est le **regain actuel d'intérêt pour cette question** dans le cadre du mouvement en faveur de l'externalisation des activités informatiques décrit au chapitre 1. La plupart des grandes entreprises ont déjà intégré dans leurs choix de ressources l'examen systématique des "solutions-progiciel" disponibles sur le marché et disposent d'outils d'arbitrage élaborés. Elles développent ainsi pour leurs systèmes d'information une approche de type "assemblage de composants" : certaines "briques logicielles" sont standardisées et peuvent être achetées (les progiciels), d'autres sont spécifiques. L'enjeu est maintenant de savoir qui va développer ces briques spécifiques et qui va réaliser l'intégration des différents composants.

Nous avons retenu comme **faisant partie de notre champ d'étude** les activités suivantes :

- le développement de nouvelles applications spécifiques,
- les nouveaux développements spécifiques portant sur des applications anciennes (maintenance évolutive, refonte d'application),
- les travaux d'intégration de type développement d'interfaces,
- les travaux d'intégration et les compléments autour de progiciel (interfaces, développements complémentaires).

C'est à l'ensemble de ces activités que nous ferons référence dans la suite de ce document à chaque fois que nous utiliserons le terme "développement d'applications". On retrouve en fait ici le domaine traditionnel d'activité des services "études" des directions informatiques internes.

Nous avons en revanche exclu de champ :

- les très petits développements de type bureautique qui sont souvent assurés par les utilisateurs eux-mêmes ou confiés à une équipe spécifique au sein de la direction informatique (critère de taille concernant les projets),

- les choix d'acquisition de progiciel, afin de ne pas confondre la question du type de ressource choisi et celle de l'origine de cette ressource,

- la maintenance "quotidienne" de vieilles chaînes spécifiques (maintenance corrective, évolutions mineures) ; nous avons en effet pensé que le problème d'externalisation pouvait se poser de façon différente pour cette activité de maintenance, les enjeux pour l'entreprise n'étant pas les mêmes concernant une application ancienne à durée de vie limitée. De plus, l'activité de maintenance est souvent séparée des autres activités de développement, tant en interne que sur le marché des services où elle fait l'objet de contrats spécifiques (dits de "tierce maintenance").

Notre champ de recherche est schématisé dans le tableau 2.2.

Portefeuille d'applications	Applications spécifiques				Applications standard
	(progiciels)				
Activités de développement	maintenances	nouveaux développements sur des applications anciennes	nouveaux développements applicatifs, interfaces	développements spécifiques autour des progiciels (intégration, compléments)	partie standard (progiciel)
Choix d'externalisation	faire en interne, faire avec assistance extérieure, faire faire				acheter



Domaine étudié

Tableau 2.2. Limites de notre champ de recherche

Ce champ de recherche a été relativement peu exploré. Les quelques travaux disponibles permettent cependant de mettre en avant des spécificités de la problématique d'externalisation appliquée à ce champ. Celles-ci concernent principalement la forme des contrats et les arguments du débat concernant le choix de ressources.

II. Les spécificités de la problématique d'externalisation appliquée au développement d'applications

2.1 La forme particulière des accords d'externalisation de l'activité de développement d'applications : le contrat par projet

Loh et Venkatraman (1992) ont cherché à comparer les différentes formes d'externalisation appliquées aux différentes activités informatiques. Ils classent les accords d'externalisation dans un schéma en fonction de deux de leurs caractéristiques : l'unité contractuelle en usage (la période ou le projet) et l'étendue du domaine organisationnel concerné (spécifique ou général).

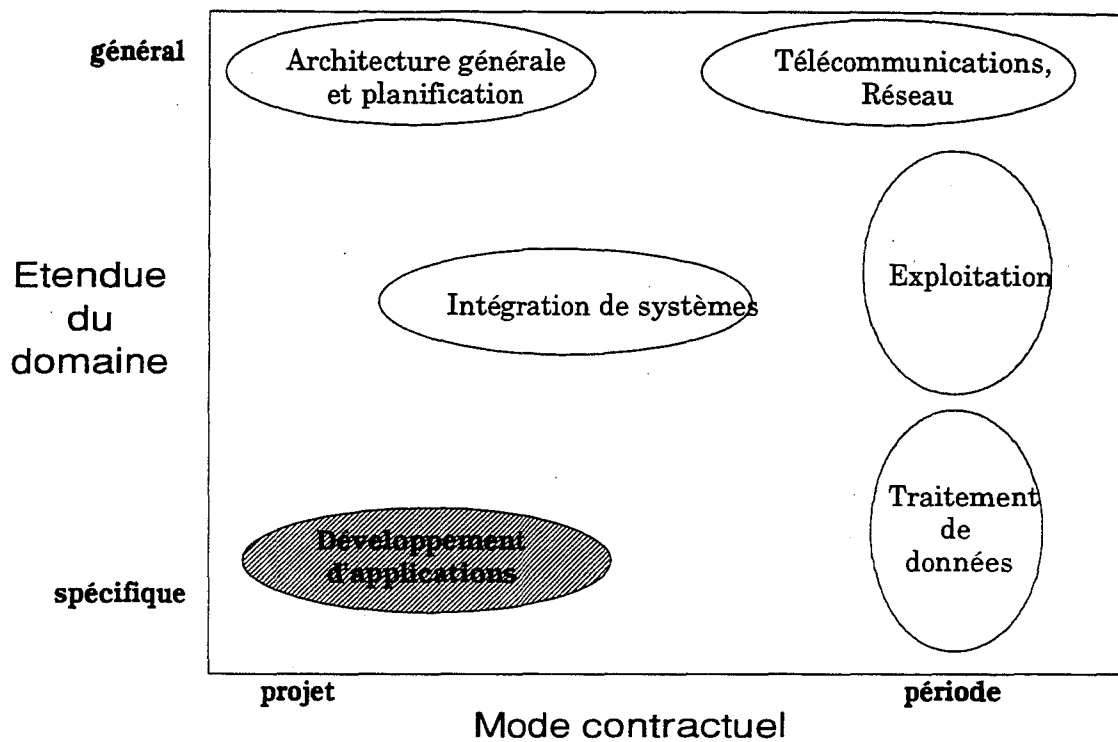


Figure 2.4. Les caractéristiques des différents modes d'outsourcing en systèmes d'information

Source : Loh et Venkatraman, (1992)

On voit ici que l'outsourcing de développement d'applications fonctionne différemment des autres formes d'externalisation. Alors que les contrats portant sur les télécommunications ou les activités d'exploitation sont généralement des contrats conclus avec un fournisseur unique, portent sur toute l'activité concernée au sein de l'organisation et définissent un accord à long terme avec un fournisseur, les accords sur le développement d'applications sont souvent conclus **dans le cadre des projets et de manière spécifique** auprès de différents fournisseurs.

Quelles sont les raisons de cette forme contractuelle particulière ? Il semble que les **caractéristiques du projet** jouent un rôle prépondérant dans la décision d'en externaliser ou non le développement. C'est en tous cas la thèse que défend Buck-Lew (1992), qui a identifié deux formes d'outsourcing d'applications et a cherché à lier ces formes d'externalisation aux caractéristiques des projets.

Buck-Lew (1992) s'intéresse au cas des grandes entreprises⁸ ayant le choix entre développement interne et appel à une ressource extérieure. Elle distingue trois modalités dans le choix de ressources :

- l'outsourcing complet consiste à faire développer complètement l'application par une équipe de projet extérieure. Le fournisseur extérieur prend la responsabilité entière du projet.

- l'outsourcing hybride consiste à faire développer l'application conjointement par une équipe de développeurs internes et externes.

- le développement interne ne fait pas appel aux ressources extérieures.

L'auteur identifie ensuite les caractéristiques des projets qui sont, selon elle, candidats aux deux formes d'externalisation (tableau 2.3).

Caractéristiques du projet	Outsourcing complet	Outsourcing hybride
TECHNOLOGIE <ul style="list-style-type: none"> • connaissance en interne de la technologie • impact du projet sur les systèmes opérationnels existants • importance de la connaissance du système d'information actuel (matériel, logiciel) 	faible faible faible	faible faible faible
GESTION DE PROJET <ul style="list-style-type: none"> • taille de l'équipe de projet • durée du projet • risque financier pour le fournisseur • taille du projet 	petite courte faible petite	toutes tailles toutes durées fort toutes tailles
CONNAISSANCE DU DOMAINE <ul style="list-style-type: none"> • connaissance interne à la fois de l'activité et des nouvelles technologies appropriées pour cette activité • importance de la connaissance des procédures et des applications existantes 	peu important peu important	peu important peu important
ORGANISATION <ul style="list-style-type: none"> • projet utilisateur • projet appuvé par la direction générale 	fort fort	fort fort

Tableau 2.3. Les projets candidats à l'outsourcing selon Buck-Lew (1992)

⁸. Il est clair que le débat sur l'outsourcing se pose surtout aux organisations ayant une taille critique suffisante pour pouvoir éventuellement gérer la ressource en interne.

Buck-Lew s'appuie sur des critères associés au projet, concernant la technologie, les risques liés à l'organisation du projet, le domaine concerné et la situation organisationnelle du décideur. L'externalisation complète serait envisageable pour de petits projets nécessitant une expertise non disponible en interne, pilotés par les utilisateurs et appuyés par la direction générale, ayant peu de liens avec les systèmes actuels et présentant peu de risques en cas d'échec pour le fournisseur.

L'externalisation partielle conviendrait à des projets de plus grande taille et présentant plus de risques pour le fournisseur. Les autres caractéristiques sont semblables à celles des projets candidats à une externalisation complète. L'auteur précise que ces projets sont souvent proposés en sous-traitance complète et que c'est à la demande du fournisseur qu'une forme intermédiaire est finalement retenue.

Les conclusions de Buck-Lew conduisent à une utilisation assez restrictive de l'externalisation, du moins sous sa forme radicale : deux critères lui semblent rédhibitoires, la taille du projet, qui doit rester faible, et le risque pour le fournisseur (dommages et intérêts potentiels en cas de mauvais achèvement), celui-ci n'étant pas prêt à s'engager dans une obligation de résultats forte. Il faut souligner cependant ici les limites de cette analyse. L'auteur n'a pas cherché à vérifier empiriquement l'application de son modèle à une base importante de projets et se contente d'illustrer son raisonnement par un exemple de chaque catégorie de projet.

2.2 Les arguments du débat sur l'externalisation du développement d'applications : une dissonance avec le discours général

Lors de notre analyse du discours sur l'externalisation des fonctions informatiques, nous avons été frappés par une **dissonance** entre le discours général sur l'externalisation de la fonction informatique (relativement favorable, sous certaines conditions) et le discours spécifique concernant l'externalisation de l'activité de développement d'applications (apparemment plus réticent). Nous montrons ici quelques exemples des réactions exprimées :

"Bud Kilsby [responsable des systèmes d'information] estime que l'entreprise n'a pas externalisé le développement d'applications parce que cette activité, contrairement à l'exploitation du centre de calcul, est considérée comme stratégique et l'entreprise ne veut pas en perdre le contrôle" (Analyse du contrat d'externalisation passé entre Jorgensen Co. et Digital Equipment Corp., IS Analyser, Septembre 1993, p.10, traduit par nous).

"La solution la mieux adaptée dans certains cas peut être la prise en charge de l'exploitation d'un système (FM d'exploitation) alors que les études et le développement des applications jugées stratégiques restent en interne" (Bancatique, Dossier Facilities Management, p.120, février 1992).

"Les discussions s'échauffaient sur les deux problèmes les plus sensibles : le maintien du software des systèmes existants et le développement de nouvelles applications. Un tiers des membres du conseil technique⁹ considérait que Continental devait contrôler le maintien des applications bancaires qu'elle achetait à l'extérieur. Un autre tiers a insisté pour que la banque conserve le développement de ses nouvelles applications maison. (Décision d'externalisation des activités informatiques prise par la banque Continental, Huber, 1993)

"Dans sa globalité, l'informatique est toujours considérée comme stratégique pour une entreprise. A ce titre, l'entreprise veut conserver, via sa direction des systèmes d'information, la maîtrise directe de tous ses développements applicatifs, et de tout ce qui différencie son système d'information" (Laigle, 1994)

"Les participants s'accordent à penser que la conception même du système d'information et les études applicatives correspondantes ne peuvent être externalisées sans risquer de perdre la maîtrise d'un élément vital pour l'entreprise". (Cigref, Evolution de la fonction informatique, 1993).

"La tendance est à davantage de réalisations en interne. La sous-traitance, si elle permet de minimiser les investissements, est devenue trop lourde à cause des frais de gestion, commerciaux et de structure. De plus, les informaticiens venant de l'extérieur manquent souvent de compétences, ignorant la nature de nos nouveaux projets, notre stratégie

⁹. Composé d'utilisateurs issus de toutes les activités.

d'entreprise. Parallèlement, nous nous sommes rendu compte qu'il était parfois plus rentable de développer certaines applications en interne, pourvu qu'elles soient pérennes" (L-N. Joly, directeur en charge de l'informatique à la Société Générale, dans 01 Références, Octobre 1992, p.115).

"Les SSII détournent le génie logiciel à leur profit : les sociétés de service commencent à utiliser les outils de génie logiciel pour améliorer la qualité et la rapidité de développement d'applications. Mais pas encore pour faire baisser la facture de leurs clients." (Le Monde Informatique, 26 novembre 1993, p.34).

Cette réticence apparente n'empêche pas certaines entreprises de "sauter le pas", loin de là :

- la banque Continental (voir exemple ci-dessus) a finalement choisi d'externaliser également de développement d'applications,

- parmi les 8 cas d'entreprise ayant signé un contrat d'outsourcing présentés dans leur ouvrage par Lacity et Hirschheim (1993), cinq concernent l'ensemble des fonctions informatiques, développements inclus,

- d'après une enquête menée fin 1993 par CSC Index auprès de 780 entreprises européennes et américaines, 66% des entreprises américaines consultées (et 61% des entreprises européennes) "envisageraient d'outsourcer en 1994 le développement d'applications", ce poste arrivant en tête des intentions d'externalisation devant la logistique et la maintenance du parc micro, les réseaux et télécommunications et bien avant l'exploitation du centre informatique (45% des réponses européennes, 31% des réponses américaines).

Ces différents exemples mettent cependant en lumière trois questions principales, spécifiques à l'activité de développement d'applications :

1. l'externalisation peut-elle permettre de réaliser dans le cadre du développement d'applications les **mêmes réductions de coût de production** que celles promises (et parfois atteintes) pour les activités d'exploitation ?

Ceci revient à poser la question de l'existence d'un réel "avantage à la spécialisation industrielle" en ce domaine : après tout, n'est-il pas plus important pour limiter les coûts de développement de connaître l'entreprise, sa stratégie, son métier que d'être spécialisé dans une technique de développement ? De plus, dans quelle mesure le sous-traitant possède-t-il un réel avantage technique sur les équipes internes ?

2. cette activité n'est-elle pas **trop stratégique** pour être externalisée ?

Nous avons pu mettre en évidence dans le chapitre 1 le lien fort qui lie la banalisation de la ressource informatique et le choix d'externalisation : si l'informatique est un produit industriel homogène et banalisé (- et c'est la thèse qui s'exprime sous la forme : "*IS is a utility*"), alors il est cohérent d'appliquer pour cette ressource un schéma de spécialisation industrielle et d'axer le choix entre ressource interne et externe sur le critère du coût. Dans le cas du développement d'applications, en revanche, un autre facteur semble devoir être pris en compte, à savoir l'importance stratégique de la fonction pour l'entreprise, qui viendrait infirmer -pour ce cas précis et si cette observation est vérifiée- l'hypothèse générale de banalisation de la ressource informatique.

3. **la relation avec des fournisseurs extérieurs** ne risque-t-elle pas de générer des **coûts supplémentaires** ?

La préoccupation concerne ici le coût de gestion de la relation client-fournisseur et l'attitude de SSII, plus propres à saisir toute opportunité d'augmenter la facturation qu'à répercuter sur le client une amélioration de la productivité.

Conclusion

Nous avons défini dans ce chapitre les limites de notre champ d'étude sur deux aspects :

- le **domaine étudié** est "l'activité de développement d'applications", qui intègre le développement de nouvelles applications spécifiques, les nouveaux développements spécifiques portant sur des applications anciennes (maintenance évolutive, refonte), les travaux d'intégration de type développement d'interfaces, les travaux d'intégration et les développements complémentaires autour de progiciels. En sont exclus les petits développements bureautique et la maintenance courante d'anciennes applications.

- le **choix d'externalisation étudié** est l'arbitrage entre faire en interne, faire en interne avec une assistance extérieure, faire faire par un fournisseur extérieur. L'acquisition de progiciel se situe donc en dehors du champ de l'étude.

Nous avons ensuite étudié les spécificités de la problématique d'externalisation appliquée à ce champ précis, qui la différencient de la problématique générale d'externalisation présentée dans le premier chapitre. Le premier point concerne la forme contractuelle des accords : l'unité de base est le **projet**, correspondant à la réalisation d'une application précise. De plus, selon certains auteurs, les caractéristiques du projet (technologie, taille, domaine, origine) joueraient un rôle important dans la décision d'externalisation.

Deuxième élément, l'externalisation du développement d'applications semble générer des interrogations particulières, à la fois :

- opérationnelles (les fournisseurs ont-ils un réel "avantage à la spécialisation industrielle" en ce domaine ?)

- stratégiques (le développement d'application n'est-il pas trop stratégique pour être externalisé ?)

- de contrôle (le coût de gestion de la relation client/fournisseur n'est-il pas prohibitif ?).

**Seconde partie :
étude analytique des approches théoriques**

L'objectif de cette seconde partie est l'élaboration du modèle et des hypothèses de la recherche empirique, au travers de l'étude analytique des différentes approches théoriques du problème.

L'examen montre que le thème de l'externalisation est très présent dans la littérature de gestion, sous différents vocables qui traduisent des époques et des approches différentes : on parle ainsi de théorie des coûts comparatifs, faire ou faire faire, de sous-traitance, d'outsourcing stratégique, d'appel à différentes structures de gouvernance (hiérarchie, marché, formes hybrides)... Dans tous les cas, la question revient à déterminer quelle partie de son activité la firme doit assurer en interne, et quels sont les éléments, produits ou fonctions qu'elle a intérêt à acquérir auprès d'un fournisseur extérieur.

Nous avons identifié quatre grandes approches théoriques rationnelles des choix d'externalisation, susceptibles de nous aider à répondre aux questions de la recherche. Nous consacrons à chacune d'entre elles un chapitre dans cette seconde partie.

1. L'approche industrielle, centrée sur les avantages de la spécialisation (chapitre 3)

La question du faire ou faire faire correspond ici à la recherche d'une optimisation du processus et du coût de production. Les concepts développés dans cette approche ont été largement diffusés : économies d'échelle, effet d'apprentissage, avantages de la spécialisation. Ils font partie des "outils de base" du management et qu'ils constituent une référence que tout étudiant en gestion et tout professionnel d'entreprise connaît. Appliquée au cas particulier du développement d'applications, cette approche nous conduit à nous interroger sur l'évolution du processus de développement et des technologies et leur impact sur les choix d'externalisation ; nous espérons y trouver un guide pour la réponse à la question : "les fournisseurs ont-ils un réel avantage à la spécialisation industrielle en ce domaine ?"

2. L'approche stratégique (chapitre 4)

De nombreux auteurs en stratégie ont insisté sur l'aspect stratégique du choix de l'origine de la ressource ; en particulier s'est développée sous le nom de "théorie de la ressource" une branche de la stratégie qui place l'externalisation au coeur de ses réflexions. Nous sommes ainsi

amenés à nous interroger sur l'intérêt stratégique de la ressource de développement d'applications, pour dégager des éléments de réponse à la question : "le développement d'applications n'est-il pas trop stratégique pour être externalisé ?"

Les deux chapitres suivants sont consacrés aux théories contractuelles ; c'est en ce domaine que la recherche sur les choix d'externalisation a été la plus féconde dans les dernières années. Ils traitent tous les deux, chacun sous un angle particulier, de la question : "le coût de gestion de la relation client/fournisseur n'est-il pas prohibitif ?"

3. La théorie de l'agence (chapitre 5) nous fournit un cadre d'analyse extrêmement précieux de la situation contractuelle et permet de mettre en évidence les risques et coûts associés. C'est également une théorie du contrôle, qui permet de définir les deux grands modes de pilotage d'une relation contractuelle.

4. La théorie des coûts de transaction s'est quant à elle imposée depuis une quinzaine d'années comme un cadre théorique exceptionnel pour étudier les arbitrages entre les différentes "structures de gouvernance" que sont le marché, la hiérarchie (l'organisation interne) et les formes intermédiaires. Nous détaillons dans le chapitre 6 les différents aspects de cette théorie, notamment les plus récents concernant l'étude des caractéristiques des formes de gouvernance. Ceci nous conduit en particulier à nous interroger sur les conditions de mise en oeuvre d'une bonne coopération entre utilisateurs et informaticiens internes, ainsi qu'à identifier les attributs de la transaction susceptibles d'intervenir sur les choix.

Le dernier chapitre de cette partie (chapitre 7) est consacré à la synthèse des différentes approches ainsi qu'à l'élaboration du modèle et des hypothèses de la recherche empirique.

Avant d'entrer dans une description détaillée des approches théoriques, nous allons préciser en quelques points les grandes lignes retenues pour l'étude analytique.

1. Nous avons cherché dans chacun des quatre chapitres théoriques à mettre en évidence un angle d'attaque particulier du problème d'externalisation, en privilégiant une vision schématique de chaque approche, centrée sur ses spécificités. Ce parti-pris a été adopté pour

structurer l'exposé et en favoriser la clarté. En réalité, ces approches, même si elles ont chacune leur spécificité, ne sont pas exclusives. Par exemple, il serait tout-à-fait inexact de dire que l'approche industrielle ne prend pas en compte les aspects stratégiques ou ignore les problèmes de contrôle de l'activité ; nous avons cependant limité la description à ce qui nous a semblé être son angle de vue particulier du problème d'externalisation, à savoir l'optimisation du processus de production.

2. Nous nous sommes volontairement placés dans une optique rationnelle des choix, ce qui nous a conduit à écarter les approches politiques comme celle de Pfeffer (1981), utilisée par Lacity et Hirschheim (1993) pour interpréter les décisions d'externalisation dans différentes organisations. Nous reconnaissons le rôle que peuvent jouer les aspects politiques dans toute décision prise par une organisation, mais avons voulu limiter notre objectif dans ce travail exploratoire à l'examen des critères rationnels influant sur la décision, en formant l'hypothèse implicite que les jeux de pouvoir ne peuvent suffire à masquer les déterminants rationnels du choix.

3. Notre principal objectif est la confrontation de chaque approche théorique à notre domaine d'application. L'exposé des grandes lignes des approches théoriques a été volontairement restreint aux seuls éléments directement susceptibles d'éclairer notre recherche. L'essentiel de l'exposé concerne donc l'application au cas précis du développement d'applications informatiques.

Chapitre 3 :
**L'approche industrielle - analyse du processus de
développement sous l'angle technologique**

Introduction

Le principal avantage mis en avant par les partisans de l'externalisation est la baisse des coûts de production ; certains ajoutent également que le recours à un fournisseur extérieur permet améliorer la qualité et les délais. Peut-on appliquer ces raisonnements au développement d'applications ?

Les arguments en faveur de la spécialisation industrielle sont bien connus ; nous nous limiterons à en faire un bref rappel dans la première section de ce chapitre. L'objectif est ici davantage d'analyser le contenu de la ressource "développement", en termes de processus, de savoir et de savoir-faire. Cela implique de mener un examen attentif et critique du "mode de fabrication" d'une application, d'isoler les points sensibles du processus de développement et d'identifier les ressources attachées à ces points sensibles.

Il faut souligner que, adoptant une perspective industrielle, nous nous intéressons ici au processus de production sous son angle technologique. Nous cherchons à faire le point sur l'état présent de la technologie puis à déterminer les problèmes et les interrogations qui se posent dans la gestion de cette technologie, ce afin d'identifier l'ensemble des éléments qui pourraient conduire les entreprises à modifier leurs pratiques.

Section 1 - Le modèle industriel : l'optimisation du processus et des coûts de production

Dans ce modèle, la question de faire en interne ou de sous-traiter correspond principalement à l'optimisation du processus et des coûts de production. Les concepts développés dans cette approche trouvent leur origine dans l'école économique libérale ; ils ont été largement diffusés. Nous allons en faire une revue rapide en nous appuyant principalement sur les travaux de Barreyre (1968) et de Dussauge et Ramanantsoa (1987), avant de décrire les principales formes que peut prendre le recours à la ressource extérieure dans ce modèle.

I. Les avantages de la division du travail et de la spécialisation

Adam Smith¹ déclarait dans Recherches sur la nature et les causes de la richesse des nations (1776) : "Les plus grandes améliorations de la puissance productive du travail et la plus grande partie de l'habileté, de l'adresse et de l'intelligence avec laquelle il est dirigé ou s'applique, sont dues, à ce qu'il me semble, à la division du travail."

C'est ce même concept de division du travail qui conduit dans une perspective industrielle à l'externalisation de certaines tâches par les entreprises : "Du principe de la division du travail à celui de la spécialisation des entreprises, il n'y avait qu'un pas, que l'école libérale a franchi aisément : les disciples et successeurs d'Adam Smith allaient faire ressortir les avantages économiques résultant de l'allongement des séries et de l'accroissement des dimensions de l'unité de production" (Barreyre, 1968, p.111).

La spécialisation permet tout à la fois d'atteindre la taille suffisante pour bénéficier de baisses du coût unitaire de production, d'améliorer la qualité de la production et de limiter les investissements technologiques de l'entreprise à quelques domaines choisis.

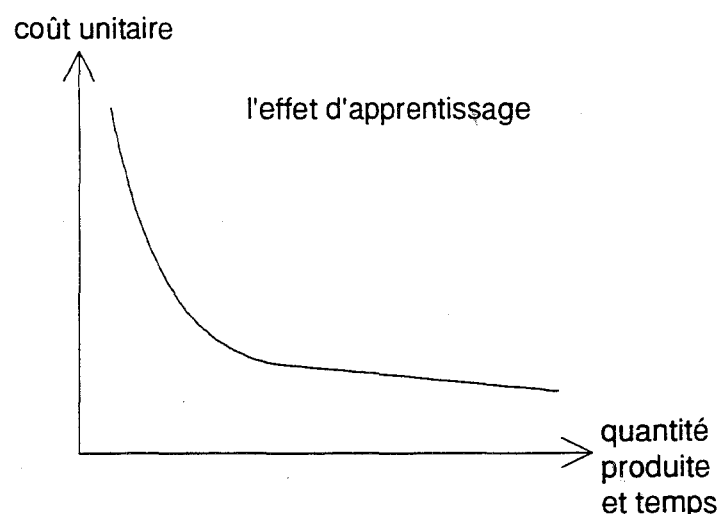
¹Cité par P.Y. Barreyre, (1968)

1.1 La relation entre coût unitaire de production et volume de production

On constate empiriquement dans beaucoup d'activités, en particulier industrielles, que le coût unitaire a tendance à décroître au fur et à mesure que le **volume de production** augmente. Ce phénomène a été analysé et a donné lieu à un concept bien connu, la "courbe d'expérience" dont les effets sont dus à trois catégories de causes (Dussauge et Ramanantsoa, 1987, p. 73) :

- l'effet d'apprentissage,
- les économies d'échelle et l'effet de taille,
- l'innovation.

a) L'effet d'apprentissage prend en compte la difficulté de maîtriser une nouvelle activité (ou une nouvelle technique au sein de cette activité) et la valeur ajoutée de l'analyse des échecs ou des défauts des premières expériences. Après un certain temps, ou un certain nombre d'essais, une activité devient familière à l'entreprise et profite de l'effet d'apprentissage, ce qui permet d'évaluer correctement le risque, de mieux cibler les objectifs et d'optimiser les moyens engagés pour les atteindre à moindre coût. Le facteur humain (stabilité des équipes) est prédominant. L'effet d'apprentissage est surtout sensible dans les premières étapes de la mise en oeuvre de l'activité et n'est donc pas systématiquement lié à des volumes de production élevés. Il peut intervenir selon la nature de l'activité et sa complexité après quelques expériences.

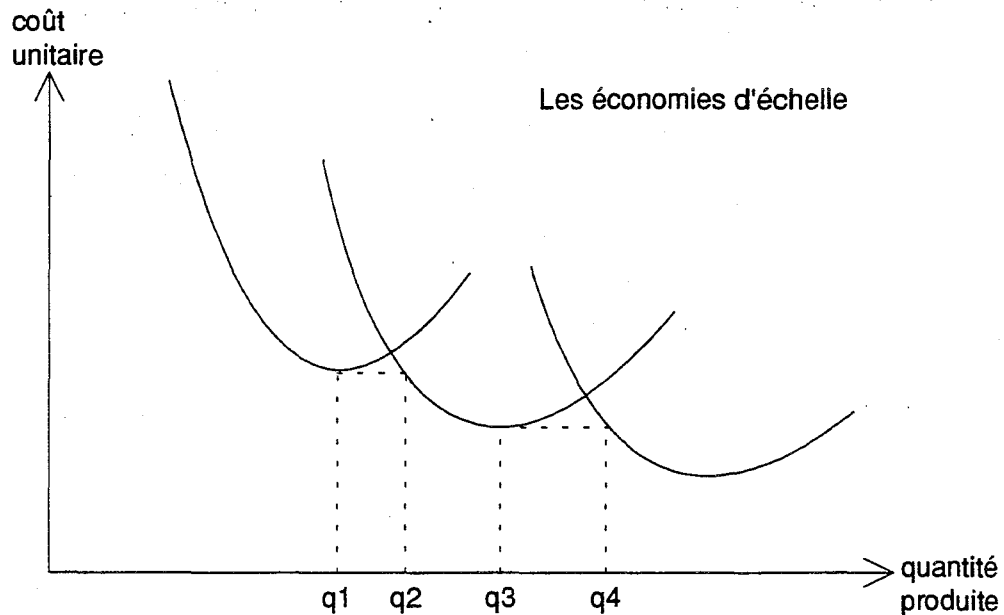


Source : Martinet (1983)

Dans le cas des logiciels applicatifs, l'effet d'apprentissage a été maintes fois constaté : on ne peut ainsi tirer partie d'une nouvelle technique (langage, base de données, AGL) que lorsque celle-ci a été expérimentée sur plusieurs projets. Par ailleurs, le coût d'apprentissage est relativement élevé, ce qui explique une certaine spécialisation du travail chez les informaticiens, très visible lorsque l'on examine les offres d'emploi (on parlera ainsi non d'un spécialiste en bases de données, mais plus précisément d'un spécialiste en bases de données relationnelles, voire d'un "spécialiste Oracle" ou "Ingres").

b) les économies d'échelle, *a contrario*, jouent en faveur de forts volumes de production. Celles-ci s'expliquent par la conjonction de deux phénomènes : l'accroissement des séries permet d'adopter des modes de production plus efficaces et d'améliorer la productivité. Mais surtout, il permet de répartir les frais fixes sur un plus grand nombre d'articles. Ce facteur est déterminant pour des activités exigeant des investissements lourds, qu'il s'agisse d'actifs physiques (appareil de production) ou immatériels (recherche et développement, technologie, marque). Dans ce type d'activité, l'entreprise ne peut espérer amortir ses frais fixes en dessous d'un certain volume de production qu'elle doit pouvoir atteindre pour avoir intérêt à produire elle-même.

La **taille de l'unité de production** et le **taux d'occupation de la capacité** sont les facteurs déterminant les économies d'échelle. La plupart des investissements sont effectués par palier, ce qui explique l'existence de plusieurs courbes de coût. A un volume de production donné, l'entreprise pourra avoir intérêt à sous-traiter une partie de sa production plutôt que d'engager un nouvel investissement, si elle n'est pas sûre d'atteindre un volume de quantités produites suffisant pour changer de courbe de coût unitaire (ainsi, on aura intérêt à sous-traiter entre q_1 et q_2 ou encore entre q_3 et q_4).



source : Barreyre (1968)

c) Les innovations apportées au produit ou au processus de production et rendues possibles par l'accumulation d'expérience sont une cause de l'effet d'expérience (Dussauge et Ramanantsoa, 1987). Cependant, pour que l'innovation soit à l'origine d'un avantage stratégique durable, il est nécessaire que la firme puisse s'en réserver pendant un certain temps le bénéfice exclusif. Cela implique en règle générale d'une part qu'elle soit à l'origine de l'invention, d'autre part que celle-ci puisse être protégée par brevet. Il faut également noter que l'innovation peut avoir pour effet de "casser" la courbe d'expérience en rendant inutile (voire nuisible, du fait de la résistance au changement) l'expérience acquise précédemment.

1.2 Les autres avantages de la spécialisation

Au-delà de la réduction unitaire du coût de production, la spécialisation industrielle présente de nombreux avantages :

- l'expérience accumulée par le spécialiste de la ressource lui permet non seulement de réduire ses coûts de production, mais d'améliorer la qualité de ses produits et de réduire ses délais de production ; la réflexion industrielle a considérablement évolué dans les dernières

années dans la maîtrise de ces deux facteurs, si bien que l'on peut maintenant considérer que la performance industrielle se mesure par un indicateur triple : coût - qualité - délai².

- dans le cas d'une technologie complexe et évolutive, elle permet d'éviter aux entreprises d'avoir à investir individuellement dans la constitution d'une expertise coûteuse à constituer et à entretenir ; seuls les acteurs spécialisés du marché investissent dans la technologie ; ils peuvent de plus le faire de façon sélective en exploitant des niches technologiques.

- les besoins de l'entreprise peuvent être variables dans le temps ; si elle utilise de façon irrégulière une expertise technologique, elle a tout intérêt à flexibiliser la ressource en faisant appel à l'extérieur.

1.3. Le calcul économique : analyse comparative des coûts

Comment calculer l'intérêt économique de l'externalisation ? Barreyre (1968) fait une synthèse des méthodes de comparaison des coûts généralement utilisées pour établir la rentabilité économique des choix. Il distingue deux grandes approches.

1- la comparaison des prix de revient

Cette méthode revient à comparer d'un côté le prix proposé par les fournisseurs extérieurs et de l'autre le prix de revient complet du bien obtenu dans l'organisation. Une variante consiste à ne prendre en compte que le coût direct et à pratiquer une analyse marginale.

Comme nous l'avons vu, cette analyse comparative des prix de revient met généralement en évidence les avantages de la spécialisation, en particulier l'amélioration de la productivité due à l'effet d'apprentissage et aux économies d'échelle. Les deux facteurs dominants sont ici la **technologie** (requérant un apprentissage plus ou moins complexe) et le **volume de production**.

². Des études ont d'ailleurs largement montré la dépendance entre ces 3 facteurs : c'est en visant une qualité totale et en réduisant les délais de production que les réductions de coût sont obtenues.

La seule comparaison des prix de revient est cependant insuffisante. L'internalisation d'une activité correspond en effet à une immobilisation de ressources par l'entreprise ; la problématique est donc celle d'un choix d'investissement. On trouve ici une deuxième approche, centrée sur la notion de coût d'opportunité.

2- le choix d'investissement et la prise en compte de coûts d'opportunité

Placée devant un choix d'investissement, l'entreprise peut en théorie se trouver dans deux situations différentes :

- soit ses capitaux sont non limités, et elle peut alors investir dans toute activité rentable (c'est-à-dire satisfaisant aux critères de comparaison des prix de revient énoncés ci-dessus),
- soit ses capitaux sont limités (cas de loin le plus probable) et elle doit alors envisager le désinvestissement dans des activités même rentables afin de réemployer les fonds ainsi libérés dans des activités plus rentables. C'est la notion de "coût d'opportunité" qui correspond au différentiel de gains entre deux investissements possibles. L'optimisation des ressources en capital impose donc de raisonner de façon comparative, en confrontant les rentabilités respectives des différents investissements envisageables. Le critère de coût d'opportunité est donc plus restrictif que la comparaison des coûts de revient ; il augmente la gamme des activités susceptibles d'être externalisées.

En pratique, ces deux méthodes sont souvent difficiles à appliquer.

La première limite provient de la difficulté d'évaluer et de connaître des coûts de revient internes. Ainsi, les méthodes d'évaluation du prix de revient prennent généralement peu en compte les variations du taux d'occupation de la capacité interne ; des "coûts cachés" peuvent être négligés ; enfin, la définition d'un prix de revient complet comporte nécessairement une part d'arbitraire dans l'affectation des coûts indirects.

Même en supposant cet obstacle franchi, il n'est pas si simple de comparer d'une part un investissement interne, engageant l'entreprise sur le long terme, avec d'autre part un prix de marché susceptible de variations.

Enfin, dans un certain nombre de cas, la rentabilité de l'investissement sera impossible à établir, l'entreprise ne pouvant pas évaluer les gains futurs. C'est généralement le cas pour les investissements stratégiques, qui ont pour effet de favoriser les conditions de fonctionnement et d'évolution de l'entreprise à moyen et long termes et auxquels les critères de rentabilité ne peuvent être appliqués (Marjerin et Ausset, 1979).

Pour en revenir au domaine précis de notre recherche, force est de constater que l'application des ces méthodes à l'activité de développement d'applications est particulièrement difficile, les données de base du calcul économique y faisant souvent cruellement défaut !

L'investissement interne a donc souvent été justifié à partir de deux critères simples :

- Premièrement, la pérennité du besoin et l'existence de volumes d'activité importants.

Le contexte de "crise du logiciel" des années 1980 laissait en effet envisager une demande croissante des entreprises en logiciels applicatifs spécifiques.

- Deuxièmement, l'idée que les avantages à la spécialisation industrielle étaient réduits.

Comme nous le montrerons dans la seconde section de ce chapitre, l'activité de développement d'application était considérée davantage sous l'angle artisanal qu'industriel. L'évolution actuelle des technologies, leur complexité et leur diversité croissante sont les facteurs qui peuvent renverser cette situation en modifiant les processus de production et en imposant des investissements lourds à l'organisation.

II. Les formes d'externalisation dans l'approche industrielle

2.1. L'étendue de l'externalisation

L'organisation de la production dans l'approche industrielle implique une chaîne de responsabilités mettant en jeu un maître d'ouvrage, un maître d'oeuvre et d'éventuels sous-traitants (figure 3.1).

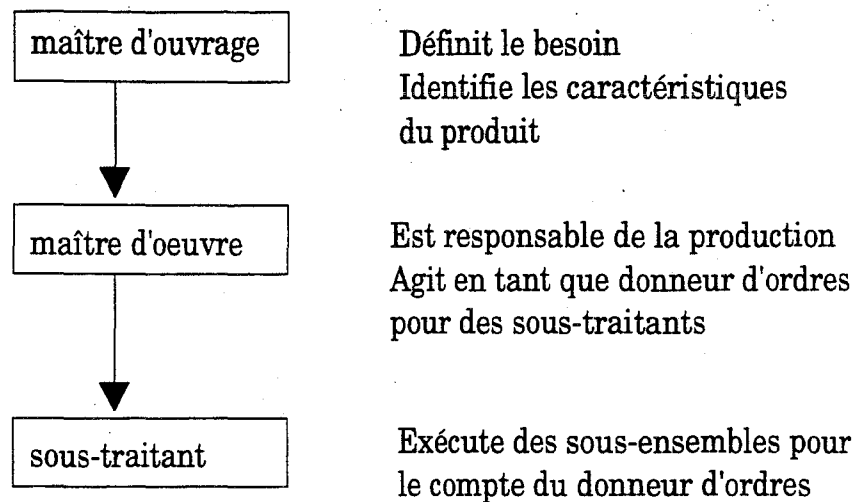


Figure 3.1. L'organisation de la production dans l'approche industrielle

En pratique, cette répartition en trois niveaux est surtout justifiée dans le cas de l'élaboration de systèmes complexes, faisant appel à différentes spécialités industrielles. Le maître d'oeuvre est alors là pour coordonner l'intervention de différents corps de métier : un exemple courant est fourni par la construction immobilière, dans laquelle l'entrepreneur responsable du chantier fait appel successivement aux entreprises de gros-oeuvre, au couvreur, à l'électricien, à l'entreprise de peinture... Dans le domaine des systèmes d'information, cette organisation correspond à l'intégration de systèmes.

A noter que cette organisation industrielle classique (maître d'ouvrage, maître d'oeuvre, sous-traitant) est peu appliquée au développement d'applications ; la définition et le contenu précis des tâches qui pourraient être menées par chacun des partenaires apparaissent encore relativement floue, bien que des efforts aient été menés pour les préciser (AFAI, 1993). Pour notre part, afin d'éviter d'introduire des confusions, nous avons choisi de ne pas utiliser la terminologie industrielle mais de préciser à chaque fois l'étape du processus (analyse du besoin, conception technique - spécification, réalisation) qui fait l'objet d'un recours à l'extérieur et d'employer le terme générique d'externalisation quelle que soit l'étape concernée (figure 3.2).

Etape du développement	Activités	Analyse de l'étendue de l'externalisation
Analyse du besoin	Aide à l'expression du besoin et à la réalisation du cahier des charges	- Utilisation du terme "Externalisation"
Spécifications	Conception technique du développement, gestion du projet informatique	- Précision à chaque fois sur : . l'étape concernée . le type d'intervention du fournisseur (main d'oeuvre, réalisation d'un sous-ensemble).
Réalisation	Programmation	

Figure 3.2. Nos règles de description de l'externalisation

2.2. Les types de sous-traitance

Pour déterminer cette revue des types d'externalisation vus sous l'angle industriel, il nous reste à rappeler la distinction classique entre sous-traitance de capacité (l'organisation n'a pas en interne la capacité de travail nécessaire en hommes, en machines - la sous-traitance répond à un problème de volume de production) et sous-traitance de spécialité (l'organisation n'a pas les compétences nécessaires - la sous-traitance répond à un problème de technicité de la ressource).

V. Malleret (1985) croise ce critère de nature de sous-traitance avec le critère de durée (appel conjoncturel ou structurel à la sous-traitance) pour définir quatre "cas-types" de sous-traitance (tableau 3.2).

DUREE

NATURE	STRUCTURELLE	CONJONCTURELLE
C A P A C I T E	<p>L'organisation ne souhaite pas investir en capacité de production pour différentes raisons :</p> <ul style="list-style-type: none"> - Développer la recherche, l'innovation - Croître plus rapidement - Garder de la souplesse. <p>Elle fait appel régulièrement à des sous-traitants qui effectuent le même travail que ceux effectués dans l'organisation.</p>	<p>L'organisation doit faire face à des pointes d'activités régulières (saisonnalité) ou non.</p> <p>Elle utilise la sous-traitance avec ou à la place d'autres moyens d'équilibre "charge-capacité" :</p> <ul style="list-style-type: none"> - Les heures supplémentaires - Le stockage - Le personnel temporaire...
T E C H N I C I T E	<p>L'organisation considère qu'elle a un certain domaine de savoir-faire qu'elle ne peut étendre à l'infini si elle veut rester efficace.</p> <p>Les activités ne relevant pas de ce domaine sont systématiquement sous-traitées.</p>	<p>L'organisation doit faire face à un besoin (interne ou en provenance des clients) dont elle ne maîtrise pas le savoir-faire.</p> <p>Elle estime que ce besoin ne sera pas durable ou qu'elle est en phase d'apprentissage et décide de le sous-traiter.</p>

Tableau 3.1. Les "4 cas-types de sous-traitance" (Malleret 1985)

III - Principaux éléments de l'approche industrielle

En conclusion sur ce rappel très rapide de l'approche industrielle de l'externalisation, nous soulignerons les points suivants :

- ce modèle s'appuie sur la capacité d'un fournisseur spécialisé à obtenir des avantages en termes de coût, délai, qualité de produit,

- les principaux éléments qui permettent d'atteindre ces avantages sont liés au processus de production ; dans ce processus de production, les **volumes** de production, la **technologie**, l'**ajustement charge/capacité** jouent un rôle dominant.

L'application de ce modèle dépend prioritairement de la nature du processus de production considéré :

-celui-ci doit être suffisamment maîtrisé et industrialisé pour permettre la constitution d'un avantage de coût par le fournisseur,

- la mise en oeuvre d'une externalisation efficace implique la possibilité pour l'entreprise de spécifier clairement le produit,

- l'appel à un fournisseur extérieur nécessite la définition de modalités de suivi et de contrôle qui sont, elles aussi, liées à la nature du processus de production.

La seconde section de ce chapitre est consacrée à l'étude du processus de développement d'applications ; nous y abordons principalement la question de l'évolution des technologies.

Section 2 - L'étude du processus de développement vu sous l'angle technologique

Bien que l'informatique soit une technique relativement jeune (mais sans doute également *parce qu'elle* est une technique jeune et donc non stabilisée) les "règles de l'art" en matière de développement ont considérablement évolué depuis l'apparition des premiers ordinateurs. La particularité de cette évolution et de cette histoire est de s'inscrire sur quelques dizaines d'années, de telle sorte que beaucoup d'informaticiens qui sont actuellement en poste dans les entreprises en ont vécu personnellement la plupart des péripéties. Les organisations ont ainsi en mémoire les réflexes acquis dans un passé récent, lors des précédentes strates d'une évolution qu'elles n'ont pas toutes menées au même rythme ni au même terme. Une synthèse préalable sur l'histoire du processus de développement des logiciels applicatifs est donc nécessaire pour comprendre l'état actuel de la technologie informatique et les conditions de sa mise en oeuvre dans les entreprises

Dans un premier temps, nous rappellerons donc quelles ont été les grandes étapes du développement des logiciels, en montrant comment les progrès ont été acquis grâce une progression parallèle des techniques et des modèles de développement. Au cours de cette évolution on a également assisté à un élargissement progressif du concept de projet informatique qui peut être défini au départ comme la résolution "en chambre" d'un problème technique et qui évolue actuellement vers la prise en compte de plus en plus automatique et rapide d'environnements évolutifs. Dans les premières étapes de l'histoire de l'informatique, le coeur du développement est la mise au point de la technique de programmation (le *how*). Mais au fur et à mesure des progrès de la technique, on cherche à appliquer l'informatique à de nouvelles classes de problèmes. On tente de passer de la résolution de cas bien définis et stables (systèmes fermés) à la prise en compte de systèmes ouverts. La difficulté majeure devient la définition même du système et de ses évolutions (le *what*) ; la technique s'oriente vers la création d'outils d'aide à la définition des problèmes et d'automatisation des changements induits par l'évolution des paramètres du système initial. Nous terminerons cette

partie par une synthèse sur les "nouveaux paradigmes de développement" (Agresti, 1986) qui marquent une rupture par rapport aux modes de production traditionnels ; nous tenterons à cette occasion d'esquisser les caractéristiques technologiques du mode de production des applications informatiques de gestion dans les prochaines années.

Dans un second temps, nous reviendrons sur les caractéristiques actuelles de la technologie de développement informatique en adoptant le point de vue des organisations qui ont à gérer cette ressource. Les nouveaux modes de production ont-ils diffusé dans les entreprises ? Dans quelle mesure celles-ci sont-elles contraintes de les adopter ? Quels sont les conditions de succès et les freins à leur mise en oeuvre ? Il nous faut donc d'abord faire le point sur le stade de développement atteint par ces nouvelles technologies, qui sont en large part d'origine américaine, et leur degré de diffusion en France. On s'aperçoit que de nombreux professionnels de l'informatique font écho d'un retard français en ce domaine. Différentes explications sont avancées. Certains auteurs mettent en avant une logique économique évoquant le passage de l'informatique de l'ère artisanale à l'ère industrielle : les investissements matériels et humains nécessaires à la réalisation d'applications seraient hors de portée d'organisations n'atteignant pas la taille critique. Pour d'autres, les différences culturelles entre la France et les Etats-Unis sont à la base de comportements différents quant à l'adoption des nouveaux modes de production. Enfin certains chercheurs ont adopté une approche organisationnelle en étudiant quelles sont les conditions en termes d'objectifs, de structure, de résultats et de système de contrôle pour que de nouvelles technologies informatiques soient assimilées par l'entreprise. Nous tenterons de montrer en synthèse de cette partie en quoi l'évolution de la technologie pose un problème *nouveau* aux entreprises et peut les inciter à une *rupture* avec leurs politiques traditionnelles de gestion de la ressource en matière de développement d'applications.

I. L'évolution des modèles de développement

Comment aborder l'histoire des technologies du développement d'application ? Nous sommes bien entendu loin de prétendre à l'exhaustivité sur un domaine aussi large. Il nous faut donc choisir une approche adaptée à notre objectif, qui est d'étudier les variations du processus de production. Les recherches de Boehm sur les modèles de développement, celles d'Agresti sur les nouveaux paradigmes, offrent une vision globale qui nous servira de fil conducteur.

1.1 B.W. Boehm et l'histoire des modèles de développement.

Boehm définit la notion de modèle de développement par ses fonctions : déterminer l'ordre des étapes du développement et définir les critères qui permettent de passer d'une étape à l'autre. Son approche est résolument orientée vers l'étude du processus de production, sans pour autant tomber dans une description pointilleuse de chaque étape du type de celle que nous proposons les différentes méthodes de projet.

Les travaux de Boehm (1984 et surtout 1988) présentent une synthèse de l'histoire du développement au travers de **six** modèles successifs depuis l'origine de l'informatique. Chaque modèle de développement propose un enchaînement des étapes du cycle de production différent, cherche à corriger les défauts du mode de production précédent et propose un nouveau champ d'application. Retraçons brièvement ces différentes étapes :

1) l'expérimentation des débuts : le modèle *code and fix*.

Les premiers développements informatiques sont caractérisés par une technique incertaine et mal maîtrisée. L'accent est mis sur la programmation, l'application se construit à travers la mise au point du programme. L'approche est expérimentale (succession de cycles très courts en deux temps : essai de programmation / détection et correction des erreurs). L'informatique peut se comparer à la résolution simultanée de problèmes mathématiques (développement de la codification binaire et de l'algorithmique) et de physiques (développement des matériaux, techniques de diffusion des signaux et de refroidissement). Les premières applications qui sont

développées selon ce modèle expérimental présentent les lacunes inévitables dans ce contexte : code peu structuré, coût des développements très élevé et surtout champ d'action très limité - on ne parle pas encore d'adéquation aux besoins des utilisateurs mais seulement d'automatisation partielle de traitements. Très vite apparaît donc la nécessité d'élargir la notion de projet au delà de la seule programmation, avec la définition de phases "amont" (analyse du besoin) et "aval" (tests, maintenance).

2) et 3) les modèles séquentiels : le cycle de la "cascade"

La structuration des étapes de développement va donner lieu à un processus de développement séquentiel (figure 3.3) qui va s'imposer comme la norme professionnelle jusqu'au début des années 1980 où l'on verra se développer à la fois des critiques structurées et de réelles alternatives techniques. Boehm distingue historiquement deux variantes de modèles séquentiels. Dans une première forme (*stagewise model*) chaque étape doit être impérativement fixée et figée avant de passer à l'étape suivante. La seconde forme (*waterfall model*, que l'on appelle également parfois *iterative waterfall model*), permet de revenir à chaque fois sur l'étape antérieure (boucles de *feed-back*) et introduit donc un peu de souplesse dans un modèle qui reste cependant très rigide.

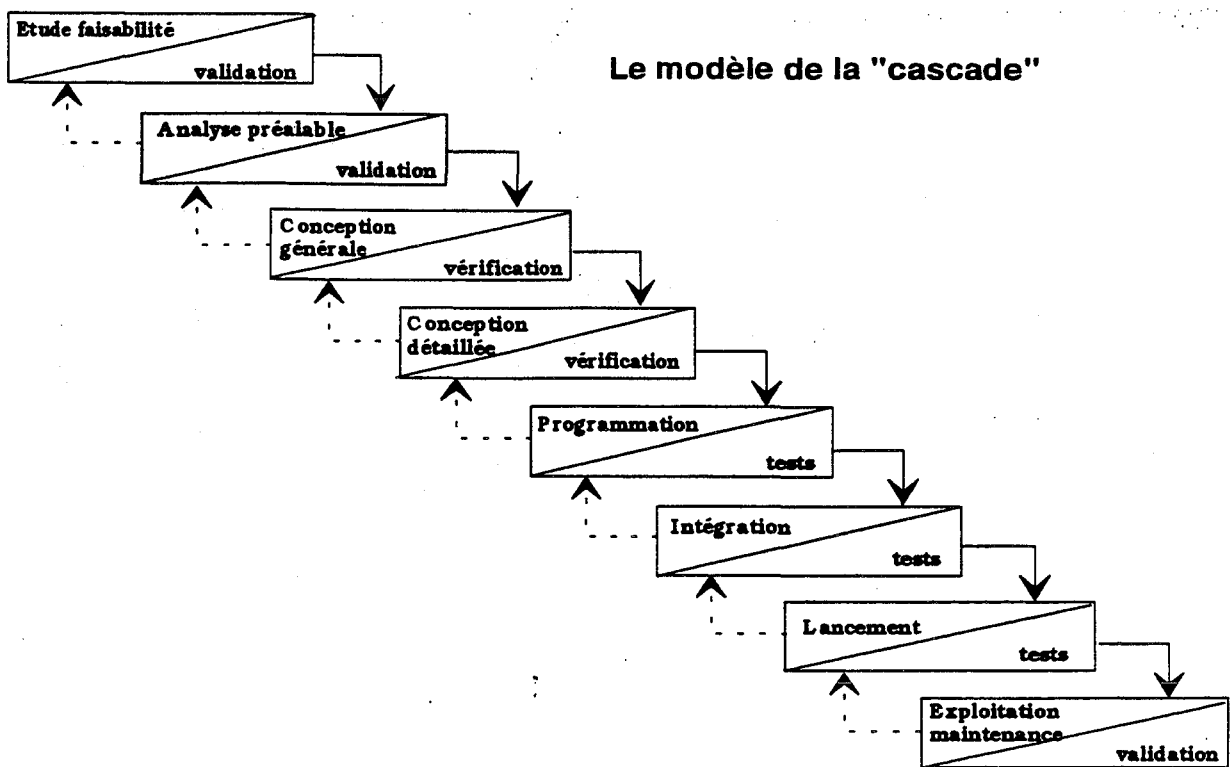


Figure 3.3. Le cycle traditionnel de développement

(source : Boehm, 1988)

Par rapport au modèle précédent, la définition des étapes s'est considérablement enrichie et surtout donne une place très importante à la phase d'analyse. Cependant fondamentalement l'orientation n'a pas changée : le coeur du projet réside dans les étapes techniques (conception détaillée, programmation, intégration). Les étapes amont d'analyse sont elles-mêmes si techniques que l'utilisateur moyen s'y perd, est souvent incapable d'exprimer ses besoins dans le formalisme voulu. Pire, il ne comprend pas le produit des étapes de conception - ou ne cherche pas à le comprendre tellement il est découragé par la lourdeur et la complexité; de ce fait, il ne parvient pas à exprimer de validation ou de désaccord motivé en cours de projet. Une incompréhension et une incapacité à communiquer s'installe entre informaticiens et non informaticiens. Le produit fini correspond plus ou moins aux besoins initiaux; si ceux-ci ont évolué en cours de route, on demande à l'utilisateur d'attendre une maintenance ultérieure. Cependant l'informatique se développe mais elle est surtout une **informatique d'automatisation de processus connus et stables** à laquelle le modèle est bien adapté.

L'évolution du contexte des entreprises, la modification du champ d'action et des objectifs de l'informatisation vont provoquer un phénomène de crise et de rejet des modèles séquentiels :

– L'environnement économique est perçu comme plus complexe et surtout plus changeant. Il est de moins en moins possible d'accepter que l'informatique continue dans une logique d'automatisation de systèmes fermés et que les objectifs et des modes de fonctionnement soient assignés à l'application dans les premières étapes du développement, sans pouvoir être remis en cause dans les étapes ultérieures. Le temps très long de développement rend cette critique encore plus aiguë.

– On attend de l'informatique qu'elle soit plus proche des utilisateurs et qu'elle couvre également le domaine de l'aide à la décision. Il faut donc inventer des moyens plus conviviaux d'expression des besoins, permettre à l'utilisateur les modifications en cours de projet, accélérer leur mise en oeuvre.

Au moment même où s'expriment de plus en plus haut ces insatisfactions (parmi beaucoup d'autres : Lucas, 1978, Freeman, 1981, Swartout et Balzer, 1982, Gremillion et Pyburn, 1983, Alavi, 1984), de nouvelles techniques permettent d'envisager une conception plus évolutive et vont révolutionner le cycle de production. Boehm identifie trois modèles faisant partie de cette "nouvelle génération", en rupture avec le cycle en cascade, apparue depuis une dizaine d'années.

4) un processus centré sur l'expression des besoins : le modèle du "développement évolutif" (*evolutionary development model*)

Si l'utilisateur ne sait comment exprimer ses besoins, et si ceux-ci sont de plus susceptibles d'une évolution rapide, alors il faut placer l'identification du problème à résoudre au coeur du cycle de production. Cette idée simple est à la base du modèle évolutif. L'objectif est de répondre aux situations dans lesquelles l'utilisateur ne sait pas définir ses besoins *a priori* ("I can't tell you what I want, but I'll know it when I see it"). Le processus consiste à construire dès la phase de spécifications un modèle pouvant être évalué et capable de simuler le comportement du système final. Cette approche, que d'autres auteurs désignent sous le nom de

"spécification opérationnelle" (Agresti, 1986), plaide pour une définition simultanée du "what" -les fonctionnalités du système- et du "how" -le mode de fonctionnement interne du système- qui sont des éléments difficilement dissociables (Swartout, Balzer, 1986). Techniquement, la phase de spécifications, contrairement au cycle traditionnel en cascade, intègre donc la définition de la structure opérationnelle du logiciel; on parle dans ce cas plutôt de *design* que de conception. En pratique, la difficulté de modifier au fur et à mesure les véritables programmes pour suivre l'évolution des besoins ainsi que la nécessité de faire évoluer ensemble des applications appartenant à un système d'information intégré limitent le recours à cette approche.

5) le modèle de la génération automatique de code (*transform model*) cherche à résoudre les difficultés de modification des programmes : dans ce modèle, le code est généré automatiquement à partir des spécifications. Bien évidemment, ceci suppose de nouveaux outils de développement capables de prendre en charge la programmation qui devient une phase "transparente". En cas de modification des besoins, les spécifications sont modifiées et de nouveaux programmes sont générés. Boehm (1988) donne comme exemple de ce modèle les tableurs (l'utilisateur ne "programme" pas sa feuille de calcul, il se contente de définir l'état qu'il cherche à obtenir) et certains langages de 4ème génération. Depuis sont apparus d'autres outils logiciels qui cherchent à assurer cette fonction de transformation.

6) le modèle de la "spirale", que Boehm présente comme le plus achevé, combine les différentes approches et propose un développement par approfondissements et extensions successifs du champ du projet. A chaque étape, une analyse de risque est faite pour la nouvelle partie du projet. Selon les risques encourus, on pourra mettre en oeuvre un développement évolutif et réaliser un prototype ou au contraire s'appuyer sur un modèle en cascade classique. Le modèle de la spirale, que préconise Boehm, n'offre pas réellement de nouveauté par rapport aux modèles existants mais s'apparente par certains aspects à un meta-modèle qui permettrait de choisir parmi les modèles précédents le plus approprié à un contexte de développement donné.

Il n'y a donc pas rejet du modèle ancien en cascade; celui-ci reste pertinent pour une certaine classe de problèmes, bien définis, très structurés et peu évolutifs. Mais en parallèle se développe avec la nouvelle génération de modèles apparus dans la décennie 80 de nouveaux paradigmes de développement (Agresti, 1986) qui révolutionnent l'industrie du logiciel.

1.2 Les nouveaux paradigmes de développement

L'excellent ouvrage d'Agresti "Les nouveaux paradigmes pour le développement des logiciels" (1986) constitue une référence majeure pour les concepts développés ci-après³. Dans une perspective qui n'est plus historique mais qui vise à faire la synthèse des grandes tendances actuelles, Agresti identifie trois paradigmes de développement, liés entre eux :

- le prototypage⁴,
- la génération automatique (*transformational paradigm*),
- la spécification opérationnelle (*operational specification*).

Les deux derniers paradigmes correspondent respectivement au *transform model* et au modèle de développement évolutif identifiés par Boehm. Cependant, contrairement à Agresti, Boehm classe dans son article de 1988 le prototypage comme l'un des derniers avatars du cycle en cascade, dans lequel le prototypage s'intégrerait sans véritablement en changer la nature : il ne s'agirait que de remplacer la phase de spécifications écrite par une phase de prototypage, plus conviviale. C'est un changement de forme et non de fond. Pour d'autres chercheurs (Nauman et Jenkins, 1982, Gladden, 1982, et donc Agresti, 1986), le prototypage marque en revanche une rupture avec le cycle conventionnel et constitue un nouveau paradigme. La structure de la phase de spécifications est profondément transformée et cette modification

³ On trouvera dans cet ouvrage un recueil d'articles ainsi qu'une bibliographie très complète sur le thème du développement du logiciel, classés par thème et commentés par l'auteur.

⁴ Le prototypage consiste à bâtir pendant la phase de conception un modèle simplifié qui permet à l'utilisateur de "voir" le produit et donc de mieux définir ses besoins.

ouvre la voie à d'autres évolutions dans les phases ultérieures du projet, en particulier la programmation et la maintenance.

Nous nous inscrivons dans cette seconde logique car le prototypage marque pour nous le **passage d'une informatique tournée vers sa propre technique à une informatique de plus en plus soucieuse de son adaptation aux évolutions de l'environnement**, ainsi que l'émergence de la nouvelle génération de processus. Dès son apparition, le prototypage génère un intérêt considérable. Ses avantages sont largement développés dans la littérature : l'usage d'un prototype améliore la phase d'intelligence du problème, facilite la communication entre utilisateurs et développeurs, permet une meilleure validation fonctionnelle, réduit le temps de tests *a posteriori*, limite le temps d'apprentissage de l'application, minimise les problèmes de rejet et permet d'élaborer des systèmes plus flexibles (Alavi, 1984, Boehm et al., 1984, Avison et Wood-Harper, 1991, Guimares et Saraph, 1992). On y voit également une solution au "goulot d'étranglement" des logiciels (Gremillion et Pyburn, 1983). Cependant, le prototypage semble être peu adapté aux problèmes très bien structurés *a priori*, pour lesquels la spécification classique du cycle en cascade reste plus performante (Boehm, 1984) ou encore aux projets posant des problèmes complexes d'optimisation technique des ressources en matériel (Guimares et Saraph, 1992).

A ses débuts, le prototype est souvent "jetable" ; une fois les besoins définis, il est remplacé par une application réalisée avec un langage moins flexible mais qui permet d'optimiser les consommations de ressources informatiques. A l'heure actuelle, Agresti considère que prototypage, spécification opérationnelle et génération automatique de code sont trois concepts étroitement liés : la spécification opérationnelle est une vision élargie du prototypage, la génération automatique de code pourrait apporter à la spécification opérationnelle la rigueur et la fiabilité qui lui font parfois défaut. Nous allons donc tenter d'esquisser, à partir de ce "trépied", la structure d'un processus de développement alternatif à celui de la cascade pour les prochaines années.

1.3. Quel processus de développement alternatif au cycle de la cascade?

A. Benveniste (1991) a proposé une formalisation de "ce qui apparaît aujourd'hui comme la méthode idéale pour la conduite de projets industriels d'automatisation", qu'il appelle "l'approche assistée". Nous avons repris son schéma initial, que nous avons adapté pour y faire apparaître les trois paradigmes d'Agresti.

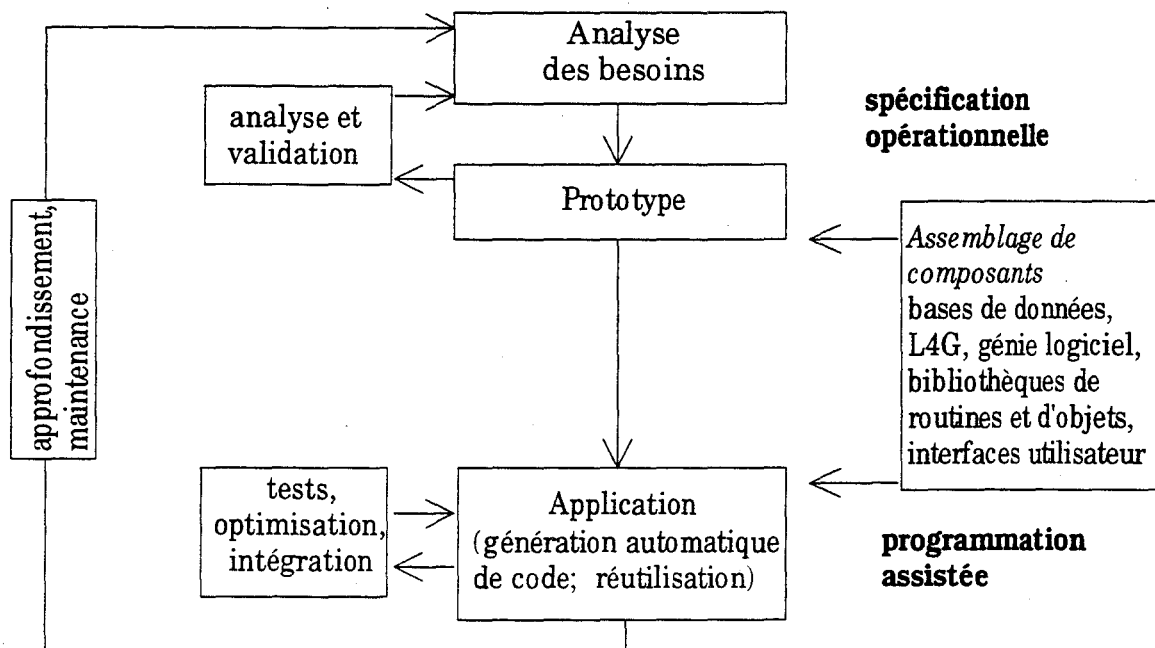


Figure 3.4. Le nouveau processus de développement alternatif au cycle en cascade
(d'après Benveniste, 1991)

Le nouveau processus de développement se caractérise par trois caractéristiques qui l'éloignent du modèle traditionnel :

1. des outils de conception et de programmation de plus en plus sophistiqués

Les processus de développement appuyés sur ces nouveaux paradigmes mettent en oeuvre des outils de plus en plus sophistiqués (Trenouth, 1991) : outils de prototypage, atelier de génie logiciel⁵, intelligence artificielle (*knowledge-based system development*), programmation

⁵. Un Atelier de Génie Logiciel (AGL) se définit comme un environnement de développement intégré rassemblant des méthodes, techniques et outils permettant le développement d'applications informatiques.

objet. Les outils de génie logiciel (*CASE-Computer Assisted Software Engineering*) ont pour objectif d'améliorer tout à la fois la qualité et la productivité du développement. La phase de spécifications, partagée entre l'utilisateur et le technicien, utilise des techniques de maquettage sur ordinateur. Celles-ci permettent, au delà de la classique définition sur papier des fonctionnalités, de définir l'interface "homme-machine" du logiciel et de tester celle-ci auprès des utilisateurs. La définition du produit fini peut ainsi déboucher sur l'élaboration d'un prototype⁶.

2. la réalisation de programmes s'apparente plus à une technique d'assemblage de code que de programmation

Le poids des "semi-finis" dans l'application finale augmente, au point que l'on peut envisager (à l'image de l'approche "objet") que la réalisation consiste en l'assemblage de briques "Lego" : certaines des briques peuvent éventuellement être produites en interne - ce sont les objets de gestion spécifiques à la firme mais **beaucoup d'éléments sont achetés auprès des professionnels du logiciel et sont standardisés** (outils d'interrogation, interface graphique). Ces briques réutilisables peuvent être testées et optimisées, produisant ainsi du code de meilleure qualité et plus efficient. La part de code programmée spécifiquement pour une nouvelle application est dans l'idéal résiduelle, correspond à l'intégration des différentes briques et à la réalisation de quelques nouveaux éléments, qui rejoignent à leur tour le dictionnaire général. Le résultat est à la fois un gain de productivité et de fiabilité.

On retrouve ici, mais à une toute autre échelle, les techniques classiques d'utilisation de sous-routines. Comme dans d'autres métiers, la standardisation des composants apparaît inévitable et permet d'augmenter considérablement la qualité du produit et la productivité du programmeur : un des exemples les plus frappants concerne toute la partie "interface utilisateur", dont on attend maintenant qu'elle soit graphique pour correspondre aux exigences de consommateurs habitués à la convivialité de l'environnement micro-informatique.

⁶. Il n'est pas toujours aisé de distinguer entre maquette et prototype : on considérera en général que la maquette se limite principalement à la définition de l'interface utilisateur (écrans, états) alors que le prototype intègre des fonctionnalités "en état de marche" et peut être dans certains cas transformable en code exécutable.

3. l'articulation classique entre les phases de conception (*what*) et de réalisation (*how*) est transformée

On retrouve bien ici, comme dans le cycle en cascade, deux grandes étapes du processus de développement : la conception - que l'on appelle plutôt "étape de spécifications"- et la réalisation (programmation). Cependant l'articulation entre ces deux étapes est profondément modifiée :

- l'étape de spécification permet à l'utilisateur de **voir le produit fini** et rend ainsi possible une évaluation du logiciel plus facile et avancée dans le temps. Il a été largement démontré que les utilisateurs éprouvent des difficultés à se plonger dans le formalisme des méthodes de conception détaillée ce qui limite fortement leur implication à ce stade du projet et repousse l'évaluation à la réception de l'application. Les nouvelles techniques de spécification à base de maquette ou de prototype sont d'une part plus intelligibles et d'autre part permettent de tester et de valider le produit à un moment du processus où les modifications sont plus faciles et les coûts engagés moins élevés. Le champ du projet peut être approfondi progressivement, suivant en cela le modèle en spirale de Boehm. La maintenance devient un cycle d'approfondissement de même nature que les autres et, au lieu d'être opérée directement sur les programmes, repart de la phase de spécifications.

- l'étape de programmation peut alors être considérée comme une **phase d'automatisation stricte**, présentant éventuellement des risques techniques mais peu de risques de divergences fonctionnelles avec le prototype validé par les utilisateurs. Surtout, le produit fini de l'étape est clairement identifié *a priori* et contrôlable *a posteriori*.

II. La gestion de la technologie informatique de développement par les entreprises

Les différents éléments que nous avons évoqués indiquent que l'industrie du logiciel, après celle du matériel, est en train de vivre un "saut technologique" qui se caractérise par la rupture à la fois des processus et des outils de production. Pour connaître les enjeux actuels, il importe

de savoir quel est le poids de ces nouvelles technologies, et à quel point elles ont pu diffuser dans les sites de production.

2.1 A quel stade de développement en sont les nouvelles technologies informatiques?

Les cabinets de conseil en stratégie distinguent différentes étapes dans le cycle de vie des technologies, en fonction de leur impact sur le jeu de la concurrence. Le modèle ADL (Arthur D. Little)⁷ identifie trois phases :

- une phase de technologie émergente, encore en développement, pendant laquelle interviennent les premières applications. La technologie a un potentiel important mais n'intervient que de façon marginale dans la mise en oeuvre de l'activité.
- une phase de technologie clé. Les technologies clé sont celles dont la maîtrise est décisive pour réussir dans l'activité et qui constituent la base de la concurrence.
- une phase de technologie de base, qui correspond à une large diffusion de la technologie et une bonne maîtrise par l'ensemble des acteurs.

Si l'on observe les caractéristiques de l'évolution technologique dans le processus de production des logiciels, un ensemble d'indicateurs montre que les nouvelles technologies ont d'ores et déjà largement dépassé le stade de l'émergence et se positionnent comme des **technologies clés** :

- les nouveaux processus et outils ont été appliqués - même si dans certains cas l'application n'a concerné que des projets restreints.
- de nombreux outils sont disponibles - même si la trop grande diversité est signe d'immaturité. Le CXP recense ainsi en 1992 près de 1000 produits composant l'offre d'atelier de génie logiciel⁸.

⁷. On trouvera une présentation de ce modèle dans l'ouvrage très complet de Dussauge et Ramanantsoa (1987).

⁸. Source : dossier Génie Logiciel, Le Monde Informatique, 7/12/92. La plupart des grandes SSII ou cabinets de conseil ont développé leur offre spécifique en AGL intégré : CGI, Arthur Andersen, Sema group, Steria, Informix.. (Dossier AGL, 01 Références, Avril 1992).

– en revanche, le stade de technologie de base n'est pas atteint et ne pourra l'être qu'à l'issue d'un processus de normalisation et de définition de standards. Une certaine normalisation intervient avec l'élaboration de standards comme PCTE (*Portable Common Tool Environment*) ou le *Repository* d'IBM mais il reste cependant difficile d'intégrer des outils différents. La diffusion est freinée par un coût d'apprentissage élevé de ces techniques⁹.

Cette évolution technologique a généré chez les professionnels et leurs clients de sérieux espoirs et de fortes attentes en termes d'amélioration de la productivité. Il paraît enfin possible de dépasser la barre des "100 lignes de programme par jour et par programmeur" qui restait depuis trente ans l'objectif non atteint. Les gains obtenus restent difficiles à chiffrer et variables selon la taille et la complexité du projet et le type d'outil utilisé. On a pu cependant identifier des rapports allant de un à trois (Tallineau, 1993), les gains étant dus autant à une meilleure maîtrise du processus qu'à l'utilisation d'outils sophistiqués.

On pourrait donc s'attendre à voir ces nouvelles techniques se diffuser dans les entreprises. Or de nombreux professionnels s'inquiètent actuellement de leur faible taux de mise en oeuvre dans les services de développement internes (Guibert, Le Gendre, Tallineau, Salzman, s'accordent pour regretter cet état de fait dans le numéro spécial de "L'informatique Professionnelle" d'avril 1993). Le prototypage se développe mais souvent sous la forme de la réalisation de modèles partiels et jetables. La réutilisation de code est encore marginale. Les outils *CASE* sont insuffisamment exploités. Le passage du développement d'application à "l'ingénierie de logiciel" s'avère extrêmement difficile à réaliser. Des réticences fortes existent, de nature technique, humaine, et organisationnelle (*I/S Analyser*, juillet 1992).

Ce problème de diffusion des nouvelles technologies peut être analysé à deux niveaux : on peut tout d'abord s'interroger sur d'éventuelles particularités du contexte français, puis se poser la question de la gestion du portefeuille technologique par les entreprises.

⁹. Le temps d'apprentissage permettant la bonne maîtrise d'un outil *CASE* intégré et par voie de conséquence une amélioration des performances d'un programmeur est évalué de 6 mois à 2 ans (*I/S Analyser*, "The costs and Benefits of CASE", Juin 1993).

2.2 Un retard français dans la diffusion des nouvelles technologies?

Sam Goldring (1993) développe une explication culturelle en comparant la situation du développement du logiciel en France et aux Etats-Unis. Cette comparaison s'appuie sur certaines caractéristiques de l'activité de développement ; les conclusions atteintes sont également avancées par des auteurs français (Le Gendre, Salzman, 1993) Nous en présentons une synthèse dans le tableau 3.3.

Pour Goldring, le développement en France se caractérise encore par un processus et une mentalité artisanales : le développement est conçu à chaque fois comme la réalisation d'une application entièrement spécifique. L'accent est mis sur une phase de conception extrêmement développée, mettant en oeuvre des méthodologies complexes. Le personnage valorisé est l'informaticien (sous-entendu le "concepteur"), dont le symbole est le Polytechnicien. Le programmeur est un exécutant et l'utilisateur n'intervient qu'en début (phase de conception) et en fin (phase de test) de processus. Les tests se confondent en partie avec la phase de mise en service de l'application¹⁰. Le système de valeurs s'appuie sur l'expertise, le savoir-faire, la recherche d'une solution élégante adaptée à chaque cas particulier. La mise en oeuvre des nouvelles technologies remet en cause la culture artisanale, va à contre courant d'une phase de conception très théorique et inquiète les programmeurs qui craignent de devenir inutiles et d'être remplacés par des générateurs de code. L'atelier de génie logiciel est davantage considéré comme un outil de productivité (menaçant l'emploi) que comme un moyen d'améliorer l'efficacité (et donc d'étendre l'utilité et l'usage des systèmes d'information). Il y aurait donc un "rejet français" des nouvelles technologies du logiciel, venant à la fois de notre culture (système de valeurs), des hiérarchies de rôle au sein des équipes de développement et de la force des traditions héritées du mode de production précédent.

¹⁰. L'expérience du lancement de SOCRATE à la SNCF, qui a perturbé pendant plusieurs mois le fonctionnement du service de réservation client, illustre entre autres problèmes cette mauvaise maîtrise de la phase de tests.

France	Etats-Unis
tendance du marché logiciel	
déclin, récession (l'âge d'or des SSII françaises serait derrière elles) coûts de production élevés	expansion, diffusion dans toutes les applications spécifiques sous formes de routines, sous-programmes coût du logiciel assez faible
caractéristiques des produits	
préférence pour le spécifique	préférence pour les produits standard
bugs acceptés, inévitables, "normaux", tests en grande partie reportés sur l'utilisateur	qualité (bug-free), les tests complets font partie du processus de production
peu d'importance accordée à la finition documentation incomplète	choix d'une interface utilisateur conviviale documentation claire et complète
caractéristiques de l'étape de spécifications	
"sur-conceptualisation": conception détaillée, très précise. méthodologies très formelles	accent sur le <i>design</i> (conception + mise en oeuvre) ; formalisation de quelques grandes idées directives pour l'application, schémas
préférence pour les grands projets. "folie des grandeurs"	préférence pour de petits projets ou des projets très modulaires
maximalisme, tendance à vouloir tout traiter. même les cas particuliers	limitation stricte du champ des projets, <i>design to cost</i>
caractéristiques de l'étape de réalisation	
accent sur la conformité aux spécifications	accent sur la créativité technique et la simplicité
réécriture complète, y compris parfois des fonctions de base (habilitations, fenêtres)	volonté de limiter le volume de code produit; utilisation de fonctions de base
les AGL sont conçus comme des outils de productivité pour diminuer le nombre des programmeurs. voire se passer d'eux	les outils <i>CASE</i> sont conçus comme un moyen d'améliorer la qualité des programmeurs et d'accélérer le flux de production
personnel informatique	
culte du concepteur : la figure emblématique de l'informatique est le Polytechnicien	culte des programmeurs ; Bill Gates déclare : "Je suis un programmeur"
séparation entre "informaticiens" (concepteurs) et programmeurs (exécutants)	pas de dissociation conception/réalisation dans les personnels des équipes
grandes équipes. grands projets	petites équipes de projet dans les premières étapes (avant tests)
système de valeurs	
valeurs "artisanales" : expertise, savoir-faire, solutions élégantes	valeurs "industrielles" : qualité, rentabilité, respect des délais

Tableau 3.2. Le développement des logiciels : une comparaison France / Etats-Unis

(D'après S.Goldring, 1993)

2.3 La gestion de la diffusion des technologies informatiques au sein de l'entreprise.

Quelle peut être l'attitude des entreprises dans ce contexte?

Pour continuer à développer du logiciel en bénéficiant des nouvelles technologies (nous excluons ici les petits développements utilisateurs sur tableurs), il faut réinvestir dans la technologie (recherche et achat d'outils), former le personnel alors même que cette technologie n'est pas solidement établie. Cet investissement s'inscrit comme nous l'avons vu dans une remise en cause du modèle technologique précédent (cycle en cascade, méthodologies de conception très formalisées, esprit artisanal) qui a été long et difficile à acquérir et dont les promoteurs ont été les équipes informatiques qui sont toujours en place. De plus, les entreprises n'ont pas fini d'absorber les évolutions précédentes de leur informatique (introduction des bases de données relationnelles, *downsizing*, réseaux) et doivent gérer des parcs applicatifs hétérogènes. Les directions générales se posent la question de l'opportunité d'un nouvel investissement et certaines directions informatiques hésitent à promouvoir un changement technologique qu'elles anticipent mal maîtriser, ce malgré les avis radicaux de certains professionnels : "Si le logiciel ne baisse pas en coût, et ne monte pas en qualité, vous serez, vous directeur informatique, un de ces jours, outsource"11.

L'explication culturelle développée par Goldring est rarement évoquée par les intéressés qui mettent en avant des arguments relevant de la rationalité économique :

- le coût d'introduction de ces technologies (acquisition des outils, formation des informaticiens). Ce problème de coût est renforcé par un contexte économique de récession défavorable aux investissements. De plus les entreprises supportent des coûts de maintenance élevés et n'accordent donc pas systématiquement la priorité aux nouveaux développements,
- l'incertitude sur la pérennité des outils - tant que la normalisation n'aura pas progressé il y a toujours le risque de se tromper de standard et de devoir réinvestir,

11. O. Le Gendre, L'informatique professionnelle, 1993.

– l'incertitude sur le caractère réellement opérationnel des nouvelles technologies. Les entreprises veulent éviter d'être les premières à tester en réel un nouvel outil; la diffusion de l'innovation est de ce fait relativement lente.

A ces différents arguments on peut ajouter l'interrogation sur l'étendue réelle des bénéfices pour l'entreprise d'une introduction rapide de ces technologies. Cash, McFarlan, McKenney et Applegate (1992) permettent de clarifier et de synthétiser les multiples éléments de ce débat sur la diffusion des nouvelles technologies informatiques en adoptant une approche organisationnelle.

S'appuyant sur le modèle de Nolan et Gibson (1974) et sur les travaux de Lewin et Schein sur le changement organisationnel, ils développent le concept de **gestion de la diffusion des technologies** informatiques (*managing technological diffusion*) et distinguent quatre phases dans l'apprentissage de la technologie par l'entreprise :

1. décision d'investissement et lancement de projets expérimentaux,
2. apprentissage et adaptation de la technologie pour gérer des tâches spécifiques,
3. rationalisation et contrôle,
4. diffusion à large échelle¹².

Nous avons repris les principales caractéristiques de ces phases dans le tableau 3.4. Les phases 1 et 2 sont celles de l'innovation, les phases 3 et 4 celles du contrôle. Les premières sont consacrées à la mise en oeuvre créative, à l'apprentissage et au test de la technologie, les secondes à sa diffusion maîtrisée.

¹². Plusieurs auteurs ont proposé des modèles de diffusion des technologies dans le domaine informatique. Ce modèle sont très proches de celui présenté ici (voir par exemple Mc Farlan, Mc Kenny et Pyburn (1983).

Phase	Condition du passage à la phase suivante
Phase 1 : décision d'investissement et lancement de projets expérimentaux	
<p>Premières formations individuelles à la nouvelle technologie. Démarrage de quelques projets avec une grande incertitude sur les coûts et les bénéfices attendus. Incertitudes et apprentissage pendant le processus de développement. Résultats des projets mitigés : dépassements de coûts importants, bénéfices peu visibles.</p>	<p>Passage en phase 2 sauf en cas de "désastre". Au quel cas, retour vers des technologies mieux maîtrisées et stagnation pendant environ deux ans, délai nécessaire pour prendre du recul et analyser l'échec.</p>
Phase 2 : apprentissage et adaptation de la technologie pour gérer des tâches spécifiques	
<p>Mise en oeuvre de projets ciblés. Apprentissage important. Résultats des projets encore mitigés et dérapages, notamment en phase d'installation. Opportunité de projets permettant le développement d'un avantage concurrentiel. Focalisation sur l'<u>efficacité</u> (atteinte des objectifs du projet).</p>	<p>Passage en phase 3 si l'apprentissage de la technologie permet de mettre en évidence son <u>utilité</u> pour l'entreprise. Sinon, stagnation en phase 2.</p>
Phase 3 : rationalisation et contrôle	
<p>Changement de perspective par rapport à la technologie : diffusion au sein d'une activité opérationnelle et surtout développement de règles précises de suivi de projet. Résultats mieux prévisibles. Moins de dérapages (coûts, délais). Objectifs d'<u>efficacité</u> et d'<u>efficience</u> (critères de coût).</p>	<p>Passage en phase 4 si l'entreprise trouve un bon équilibre entre efficacité (application de la technologie à de nouveaux projets à forts bénéfices potentiels) et efficience (rationalisation, maîtrise des coûts).</p>
Phase 4 : diffusion à large échelle	
<p>Diffusion vers d'autres activités de l'entreprise. Prévisions de résultat fiables. Maturité.</p>	

Tableau 3.3. La diffusion des technologies informatiques dans l'organisation

(D'après Cash et al. (1992))

Pour franchir avec succès les étapes de la diffusion de la technologie, l'entreprise doit disposer :

- d'une cellule de type "laboratoire de recherche" ou "observatoire de technologies" qui effectue la phase 1. L'expérimentation est limitée à de petits projets. Les contraintes de coût et de rentabilité doivent être minimales. La décision de constituer une telle cellule dépend principalement de l'**importance stratégique** des nouvelles technologies pour l'entreprise,

- d'un groupe de managers, soutenus par la direction générale qui accepte d'accorder les fonds, capables de vendre les nouvelles technologies au sein de l'entreprise, malgré l'incertitude sur l'efficacité (phase 1 et 2) et sur l'efficacité (phases 1,2 et 3). La **culture** de l'entreprise, réceptive ou non réceptive à l'innovation technologique et l'existence de liens établis entre la cellule de recherche et les opérationnels sont déterminants à ce stade.

- d'un **système de contrôle** efficace pour les technologies bien diffusées dans l'entreprise (phase 4), identifiant le champ d'action de la technologie, les objectifs à atteindre dans le cadre des projets et des critères d'évaluation de l'efficacité. Des **résultats** satisfaisants doivent être atteints pour que l'entreprise continue à investir pour renouveler son portefeuille de technologies.

On s'attendra donc à constater une bonne maîtrise des technologies informatiques dans les entreprises réunissant la plupart des conditions suivantes :

- l'informatique présente un véritable intérêt stratégique,
- les dirigeants perçoivent cet intérêt stratégique et en font un axe de leur politique,
- la culture d'entreprise est réceptive à l'innovation technologique,
- l'entreprise a atteint un bon niveau de contrôle de l'informatique classique (maîtrise de la qualité, des coûts et des délais).

Dans le cas contraire, on s'attend à trouver un comportement "suiveur" sur le plan technologique. Ce comportement suiveur présente alors le risque d'une obsolescence rapide de l'outil de production, qui ne manquera pas d'être perçue par des utilisateurs qui trouvent sur le marché des progiciels les moyens d'effectuer des comparaisons, notamment en termes

d'ergonomie. **Qu'elles choisissent ou non d'investir dans les nouvelles technologies de développement, le saut technologique oblige ainsi les entreprises à reconsidérer l'activité de production d'applications informatiques et induit la question sur la continuité ou l'interruption de cette activité en interne.** C'est sur cette "nouvelle donne" que nous allons maintenant conclure cette partie sur la gestion de la ressource technologique.

2.4 La nouvelle donne du choix des ressources technologiques

Nous avons montré dans cette section que le processus de développement suit actuellement une profonde évolution : évolution de la définition et de l'enchaînement des tâches au travers de l'apparition de nouveaux modèles de développement, évolution des techniques et outils de développement qui font passer cette activité d'un stade quasi artisanal à un stade industriel, ainsi que l'anticipait le rapport de l'OCDE, "Logiciels : l'émergence d'une industrie" (1985). La **situation** a donc objectivement changé depuis l'époque où les entreprises ont défini leurs services études informatiques (tableau 3.5). La concurrence des progiciels de gestion, que nous avons évoquée au début de notre travail, renforce la question sur l'utilité et le coût de ces services.

Il semble que l'on se trouve à un moment charnière de la gestion de la ressource technologique de développement.

– l'évolution technique intervient comme un catalyseur des interrogations des entreprises sur leur fonction de développement interne de logiciels, parce qu'elle génère de nouveaux espoirs (plus de convivialité, plus de maîtrise, des coûts moins élevés) et parce qu'elle exige de nouvelles ressources (investissements dans des outils de génie logiciel, formation). Le *statu quo* (programmation en Cobol, délais longs, implication faible d'utilisateurs dépassés par la faible convivialité des anciennes techniques de spécifications) apparaît de plus en plus difficile à tenir.

Transformation du mode de développement	nouveaux processus et outils de développement : redéfinition des étapes, séquençement différent, outils de conception et de programmation (AGL), techniques d'assemblage, réutilisation de code
Nouvelle technologie encore immature	diversité de l'offre trop importante, engendrant la confusion, normalisation partielle mais non unifiée (<i>PCTE, AD/Cycle - Repository</i>), outils encore peu éprouvés pour de gros projets (approches Objet), productivité des outils difficile à établir
Diffusion faible dans les entreprises	écart entre le discours (standardisation, réutilisation, industrialisation) et la réalité : désintérêt des directeurs informatiques pour le logiciel par rapport au matériel; coûts d'apprentissage élevés; risque de baisse de productivité des programmeurs à court terme du fait de la formation et de l'inexpérience des techniques
Contexte défavorable	poids très élevé de la maintenance des chaînes anciennes (coût, temps), contexte économique défavorable aux investissements (récession), concurrence des progiciels de gestion
Conditions de l'investissement en nouvelles technologies (Cash et al.)	<ul style="list-style-type: none"> - intérêt stratégique de l'informatique pour l'entreprise - perception de cet intérêt stratégique par les dirigeants - culture d'entreprise réceptive à l'innovation technologique - bonne maîtrise de l'informatique classique (qualité, coûts, délais)

Tableau 3.4. La nouvelle donne du choix des ressources technologiques

– il faut repenser le rôle technique des informaticiens dans les deux grandes phases du projet : pour la partie spécifications, généraliser les approches conviviales de type prototypage et aboutir le plus vite possible à un "produit fini" de la phase de spécifications qui soit une représentation suffisamment claire et complète pour que la décision d'arrêt ou de continuation du projet soit fondée. Pour la partie programmation (réalisation), développer une mentalité "d'assembleur" de logiciels, définir dans chaque projet ce qui doit être fait sur mesure et ce qui peut être acheté auprès des spécialistes du secteur informatique (outils de développement, générateurs de code, langages et bases de données, interface graphique, bases progiciels).

- à cette occasion se trouve posée la question de l'origine de la ressource de développement : la qualité, la maîtrise des coûts, la réduction des délais peuvent-ils être assurés de façon compétitive en interne? A court terme, il est clair que les éditeurs de logiciel et les sociétés de service en ingénierie et informatique possèdent une avance, en particulier pour l'étape de **réalisation**. On peut faire deux hypothèses sur le décalage actuel : on peut tout d'abord penser qu'il est temporaire et que dès que la normalisation aura fait son effet les nouvelles technologies atteindront un stade de maturité et pourront être largement diffusées dans les organisations. A l'appui de cette thèse, il faut rappeler l'ergonomie des nouveaux outils qui fait que le développement est à la fois plus simple et plus rapide qu'avec des langages classiques. Mais on peut également soutenir que "le sens de l'histoire" du développement informatique est de s'inscrire dans une évolution logique de spécialisation au fur et à mesure que les techniques s'affinent et requièrent des investissements plus importants. Dans ce cas, la technologie de développement deviendrait l'affaire exclusive d'entreprises spécialisées.

Le tableau 3.5 reprend les principales conclusions de ce chapitre. On y retrouvera en particulier les principales variables à prendre en compte dans les choix d'externalisation dans une approche industrielle :

- pérennité de la demande (sans laquelle l'investissement ne peut être justifié),
- coûts de production comparés entre l'organisation interne et le fournisseur extérieur,
- critères opérationnels d'ajustement charge/capacité et de délais,
- technologie du processus.

Le facteur technologie joue à la fois sur la gestion globale de l'activité (quels sont les investissements dans la technologie nécessaires à la poursuite de l'activité en interne) et sur les décisions opérationnelles (l'entreprise connaît-elle la technologie de réalisation de tel ou tel projet).

Principaux éléments de l'approche industrielle	Application à l'activité de développement d'applications	Difficultés et limites
<p>Cette approche est centrée sur l'optimisation du processus, la réduction des coûts de production et l'évaluation économique comparée des investissements.</p> <p>L'optimisation est obtenue grâce à la spécialisation industrielle.</p> <p>Différents types d'externalisation peuvent être envisagés. On distingue en particulier :</p> <ul style="list-style-type: none"> - la sous-traitance de capacité de la sous-traitance de spécificité, - la sous-traitance conjoncturelle de la sous-traitance structurelle. 	<p>Le processus de développement est étudié sous l'angle technologique.</p> <p>La situation actuelle se caractérise par une rupture technologique :</p> <ul style="list-style-type: none"> - transformation du mode de développement, - incertitude sur la qualité de l'offre technologique (outils logiciels), - faible diffusion des nouvelles technologies dans les entreprises. <p>Le contexte actuel est défavorable à l'adoption en interne des nouvelles technologies :</p> <ul style="list-style-type: none"> - poids élevé de la maintenance, - réduction des budgets informatiques. <p>L'externalisation peut être envisagée :</p> <ul style="list-style-type: none"> - comme un moyen d'apprentissage de nouvelles technologies (situation de transition), - comme une tendance à long terme (évolution vers une spécialisation structurelle). 	<p>L'offre technologique n'est pas parvenue à maturité ; les SSII françaises n'ont pas terminé leur période d'apprentissage.</p> <p>Le processus de production des logiciels spécifiques est imparfaitement industrialisé ; la maîtrise des coûts, de la qualité et des délais par les fournisseurs n'est pas complète.</p>

Conditions de mise en oeuvre de l'externalisation	Variables à prendre en compte dans la décision d'externalisation
<p>Identification des coûts internes.</p> <p>Définition de moyens de contrôle des sous-traitants (coût, qualité, délai).</p>	<ul style="list-style-type: none"> • pérennité de la demande • coût de production • capacité • délai • connaissance de la technologie.

Tableau 3.5 Synthèse de l'approche industrielle

Chapitre 4 :
l'approche stratégique de l'externalisation

Introduction

Nous avons vu dans le second chapitre que de nombreux professionnels et chercheurs mettent en avant l'importance stratégique forte des nouvelles applications informatiques pour l'entreprise pour justifier le maintien de cette activité en interne. L'activité de développement d'applications est-elle trop stratégique pour être externalisée ? A l'inverse, si cette activité n'est pas stratégique, faut-il l'externaliser ? Quels sont les liens entre l'importance stratégique d'une ressource et le choix du maintien de sa production en interne ou de son externalisation ? Nous allons tenter dans ce chapitre d'apporter des éléments de réponse à ces différentes questions.

La première section sera consacrée à l'analyse de **l'approche stratégique de l'externalisation**. La définition et la gestion des ressources des entreprises, le choix entre ressource interne et externalisation sont des thèmes qui sont récemment réapparus dans le champ de la littérature en stratégie. Sous le nom de "théorie de la ressource" (*Resource-based theory*) s'est développée une vision stratégique de la firme "non pas au travers de ses activités sur le marché, mais en la considérant comme une combinaison unique de ressources tangibles et intangibles" (Wernerfelt, cité par Collis, 1991, traduit par nous). Cette approche met l'accent sur les capacités spécifiques de chaque entreprise et sur ses compétences de base ; elle a pour conséquence de préconiser une gestion stratégique des ressources, un recentrage sur le métier et l'externalisation des fonctions non stratégiques.

Selon cette approche stratégique de la gestion de la ressource, il convient, pour définir les contours de l'organisation interne, de s'interroger sur les activités et les compétences génératrices de valeur ajoutée ainsi que sur les sources potentielles d'avantage concurrentiel.

Nous nous poserons donc dans la seconde section de ce chapitre la question du rôle stratégique du développement d'applications ; puis nous définirons dans la dernière section les contours d'une politique stratégique de la ressource développement d'applications.

Section 1 - L'approche stratégique de l'externalisation

L'argumentaire d'une approche stratégique du problème d'externalisation des ressources peut être analysé en trois temps :

1) plus que sa position concurrentielle actuelle sur les marchés, c'est la capacité d'une firme à identifier, à développer et à exploiter ses ressources et compétences de base génératrices de valeur ajoutée qui fonde sa compétitivité ; celles-ci doivent recevoir toute l'attention de l'entreprise et être gérées en interne ; elles forment le "coeur de son métier". Ce courant de recherches en stratégie s'est développé sous le nom de "théorie de la ressource",

2) l'entreprise doit mener une réflexion stratégique sur la justification du maintien en interne des activités secondaires, comparer pour chacune d'entre elles la performance de ses équipes internes avec celle des acteurs du marché, ce qui la conduira le plus souvent à choisir l'externalisation,

3) ce faisant, l'entreprise doit néanmoins s'assurer de conserver la disponibilité et la maîtrise de la ressource.

I. Les ressources de base de l'entreprise

Depuis quelques années, certains chercheurs en stratégie ont souligné la nécessité de rééquilibrer l'analyse stratégique - très orientée dans les années 80 vers l'étude des marchés et de la concurrence sur les produits - par une réflexion parallèle sur les compétences de l'entreprise et la gestion de ses ressources. L'originalité de ce courant de recherche, identifié sous le nom de théorie de la ressource, est "l'attention qu'il porte aux ressources internes de la firme, à la façon dont ces ressources sont achetées ou développées" (Collis, 1991).

Les ressources et les aptitudes de l'entreprise, qui sont sa première source d'avantage concurrentiel, permettent de définir les bases de sa stratégie. Grant (1991) a proposé un modèle en cinq étapes pour formaliser cette approche (figure 4.1).

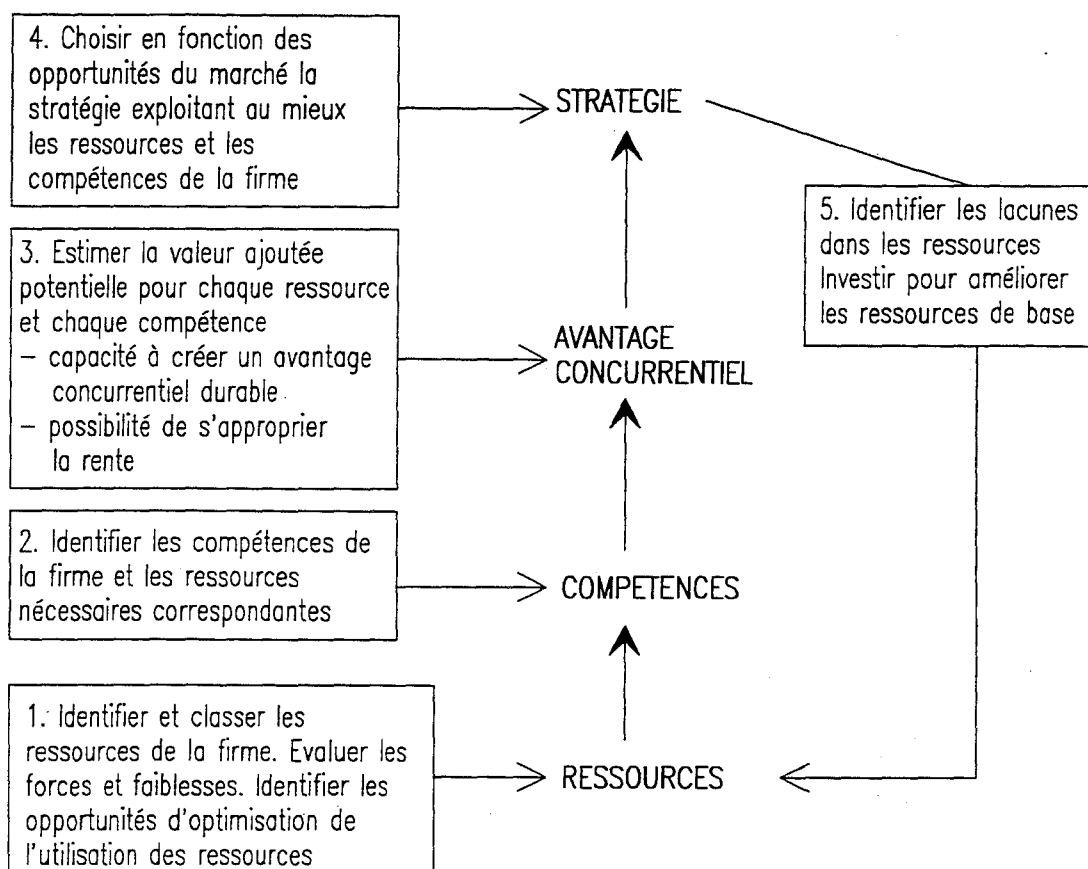


Figure 4.1. L'approche de la théorie de la ressource sur l'élaboration de la stratégie

(source Grant, 1991)

Dans cette approche, la firme va chercher à faire l'inventaire précis de ses ressources corporelles et incorporelles et se poser pour chacune d'entre elles les questions suivantes :

- quelles sont les opportunités permettant de **réduire le coût d'utilisation** de la ressource ?
On recherche l'utilisation optimale des moyens et l'amélioration de la productivité.
- quelles sont les possibilités de **valorisation des ressources existantes** ?

A travers cet inventaire et lors des étapes suivantes d'identification des compétences et des avantages concurrentiels va se dégager ce qui sera "le coeur du métier" de l'entreprise, c'est-à-dire un ensemble de compétences et de ressources vitales que l'entreprise doit impérativement gérer elle-même (Prahalad et Hamel, 1990). De façon plus précise, Grant (1991) définit ce

coeur du métier comme l'ensemble des ressources et compétences "durables, difficiles à identifier et à comprendre par les concurrents, que l'on ne peut acquérir facilement et rapidement sur le marché, sur lesquelles la firme possède un droit de propriété clair et qu'elle maîtrise bien" (p.129, traduit par nous.).

II. Les avantages de l'externalisation des activités secondaires

L'apport de la théorie de la ressource est de mettre en évidence la nécessité d'une véritable gestion à long terme des ressources de base de l'organisation. Mais pour parvenir à ce résultat, la firme doit se concentrer exclusivement sur le coeur de son métier et **savoir externaliser les activités secondaires** :

- l'entreprise ne dispose pas d'un volume de ressources illimité lui permettant de se maintenir au meilleur niveau dans toutes ses activités ; elle est donc conduite à faire des arbitrages et à privilégier certaines activités au détriment d'autres,

- n'investissant que de façon limitée dans les activités secondaires, elle devient moins efficace et moins efficiente que les prestataires de services du marché ; elle a alors intérêt à externaliser ces activités.

L'externalisation des activités secondaires est alors utilisée comme un moyen pour renforcer les ressources de base, comme le montrent très bien Quinn, Doorley et Paquette (1990) (figure 4.2) :

- en générant des **bénéfices directs**

les ressources secondaires sont obtenues de façon plus économique ; l'amélioration de la qualité de service, la possibilité d'avoir accès à un meilleur niveau d'expertise permettent d'obtenir un meilleur support pour les ressources essentielles,

- en générant des **bénéfices indirects**

l'entreprise se concentre sur ses véritables objectifs stratégiques ; l'externalisation réduit la taille de l'organisation, permet de limiter le nombre de niveaux hiérarchiques et réduit ainsi la bureaucratie interne ; de plus, l'externalisation permet d'introduire une pression concurrentielle

interne profitable à l'ensemble des activités de l'entreprise et fournit une base de référence pour les coûts.

Les auteurs soulignent que c'est la **maturité de l'offre actuelle de services** qui rend possible cette approche : "les technologies des services offrent maintenant des économies d'échelle suffisantes, une flexibilité, une efficacité et des potentiels liés à la spécialisation que les fournisseurs extérieurs peuvent exploiter pour prendre en charge beaucoup de fonctions internes à moindre coût et avec une valeur ajoutée bien supérieure" (p.79, traduit par nous).

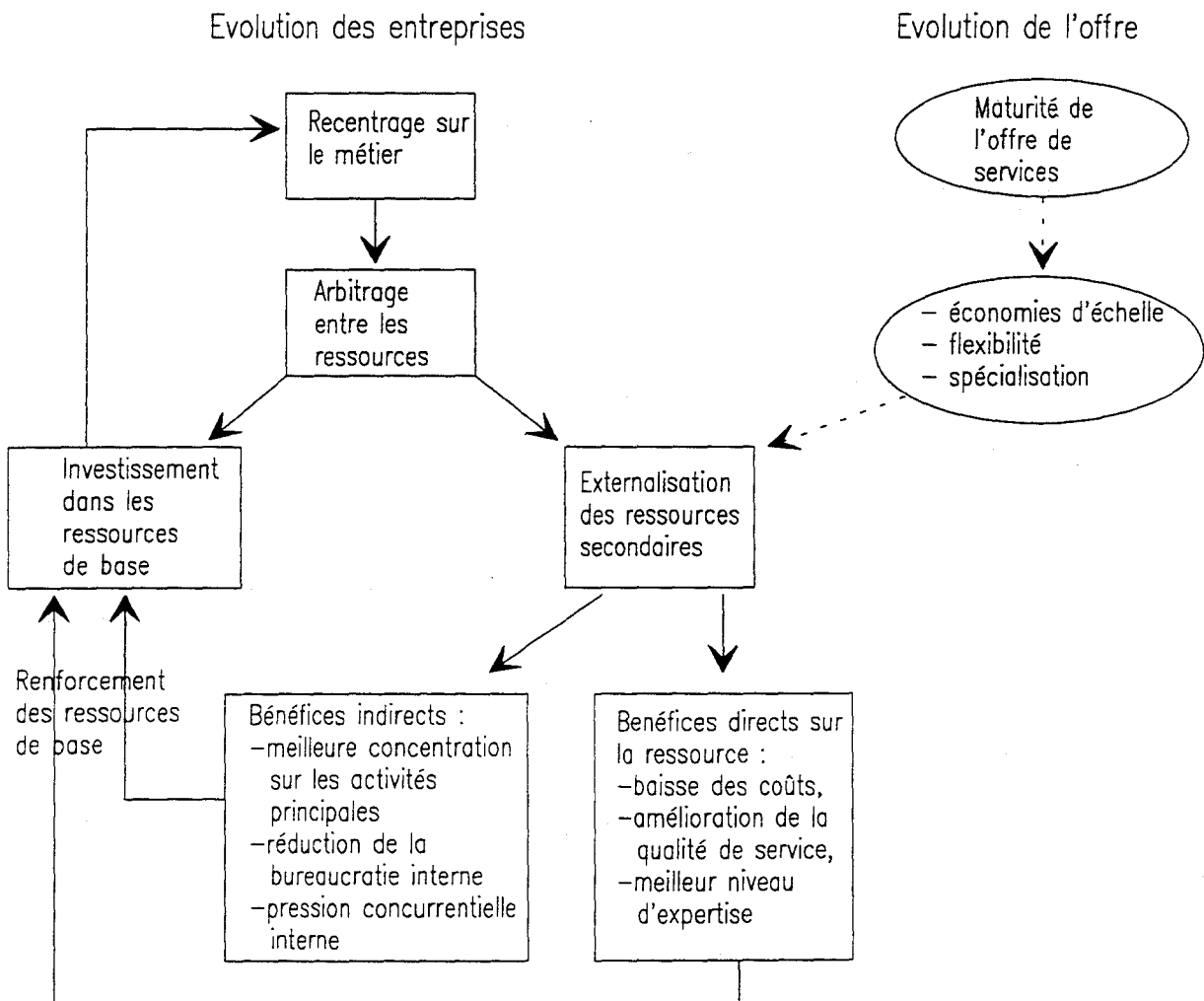


Figure 4.2. Les avantages de l'externalisation des activités secondaires dans une perspective stratégique (d'après Quinn, Doorley et Paquette, 1990)

De façon similaire, Venkatesan (1992) définit ainsi les grandes lignes d'une politique stratégique de ressources :

- focaliser l'effort de l'entreprise sur quelques ressources critiques dans lesquelles l'entreprise peut développer des compétences et un avantage concurrentiel distinctif,
- externaliser systématiquement les ressources pour lesquelles les fournisseurs extérieurs disposent d'avantages en termes d'économies d'échelle, de coût ou de qualité de prestation,
- utiliser l'externalisation comme facteur de motivation et d'amélioration de la performance pour les activités internes.

L'auteur propose ainsi d'examiner la performance des équipes internes travaillant sur des ressources non stratégiques. Si celle-ci s'avère inférieure à celle du marché, il convient alors de s'interroger sur la qualité du management de l'activité et sur la motivation des équipes et de proposer au responsable d'activité et aux employés le marché suivant : leur laisser un an à 18 mois pour amener leur niveau de performance aux standards du marché et décider l'externalisation si les résultats ne sont pas atteints. Cette politique peut sembler trop rigoureuse sur un plan social ; elle est *in fine* justifiée pour assurer la survie de l'entreprise dans des marchés de plus en plus concurrentiels : "Quand une entreprise assure en interne un service que d'autres achètent ou font produire à l'extérieur avec plus d'efficacité ou d'efficience, elle sacrifie une partie de ses capacités concurrentielles" (Quinn, Doorley, Paquette, 1990, p.82, traduit par nous).

III. Les risques de l'externalisation et les garanties nécessaires

Il reste que l'externalisation peut présenter des risques stratégiques.

Ne contrôlant plus directement la ressource en interne, l'entreprise peut craindre différentes **formes de défaillance de la part du pourvoyeur de la ressource** : écarts dans la qualité du produit fourni, mauvais respect des délais, voire dans les cas extrêmes rupture des approvisionnements suite à une faillite du fournisseur ! C'est pourquoi l'externalisation suppose

avant tout une bonne maturité de l'offre et l'existence de mécanismes de garantie sur la qualité des produits et des fournisseurs¹.

Quinn, Doorley et Paquette (1990) envisagent quant à eux comme risque stratégique principal l'établissement d'un **lien de dépendance** vis à vis du fournisseur, qui peut évoluer en une situation de domination du client par son fournisseur.

Pour éviter ce lien de dépendance, les auteurs proposent deux types d'action :

- éviter de s'adresser à une source extérieure unique ; cela suppose de mettre en concurrence systématiquement plusieurs fournisseurs et de limiter la part d'activité exercée par chacun,

- dans les cas les plus sensibles, limiter volontairement l'externalisation en choisissant de contrôler directement les phases critiques du processus.

Une troisième crainte est la **perte de savoir** au profit des fournisseurs, la politique d'externalisation systématique d'activités internes conduisant au final à une entreprise désincarnée ("*hollow corporation*"). Venkatesan (1992) rappelle que l'externalisation ne doit concerner que des activités secondaires pour lesquelles l'entreprise a une performance plutôt inférieure à celle du marché ; *ce n'est que si l'entreprise ne sait pas opérer cette distinction* dans ses ressources qu'elle risque une perte de savoir-faire. Il reste que la distinction peut s'avérer difficile à faire *a priori* et qu'un savoir perdu suite à une externalisation est long et coûteux à reconstituer.

C'est pourquoi l'auteur avance un deuxième argument en distinguant le savoir technique, qui est celui de l'exécutant de la tâche, à la "compétence architecturale" (*architectural knowledge*) : "il faut faire une distinction entre produire un sous-ensemble en interne et **contrôler** sa production et sa conception en restant expert dans ce que l'on peut appeler la compétence architecturale. Celle-ci consiste à être capable de comprendre les besoins du client et à les traduire en termes de performances à atteindre par le sous-ensemble" (Venkatesan,

¹. Nous reviendrons largement sur ces thèmes dans le chapitre 6, consacré à l'analyse des théories contractuelles.

1992, p.102, traduit par nous). Cette compétence est celle de l'assembleur qui est capable de définir précisément les caractéristiques des sous-ensembles et d'utiliser les compétences et les avantages de coût des fournisseurs : "Et donc quand il existe de bons fournisseurs pour un sous-ensemble, il n'est pas tant important d'être capable de concevoir et de produire ce sous-ensemble en interne que d'être capable d'en spécifier et d'en contrôler les caractéristiques" (p.103, traduit par nous).

Appliqué au développement d'applications, ce raisonnement nous conduit à opérer une distinction entre les différentes étapes du processus, en séparant :

- d'un côté la phase d'analyse du besoin, dans laquelle s'exerce la compétence architecturale (compréhension des besoins, traduction en termes de performances à atteindre) et qui devrait donc être maintenue en interne sous peine de perte de savoir,
- de l'autre les phases de production et d'exécution (conception technique, réalisation) qui font appel à un savoir technique et que l'entreprise pourrait envisager d'externaliser.

Section 2 - L'approche stratégique de l'externalisation appliquée au développement

I. La proposition de Rands d'un cadre d'analyse

Rands (1991b) propose un cadre normatif des choix entre ressource interne et externe axé sur la distinction classique entre le niveau stratégique et le niveau tactique (figure 4.3) :

- au niveau stratégique est déterminée la **politique** à long terme de **planification de la ressource**, qui détermine les choix internes d'investissement et de désinvestissement ainsi que l'établissement de liens avec des fournisseurs externes. Cette politique est influencée par la stratégie de l'entreprise en matière de systèmes d'information et par les conditions d'acquisition sur le marché extérieur.

- au niveau tactique sont prises les décisions opérationnelles d'allocation des ressources pour **chaque projet**, sous contrainte des capacités disponibles.

La politique de ressource va permettre de planifier la capacité globale de l'entreprise à partir des sources internes et externes, les décisions opérationnelles vont servir à gérer l'allocation de cette capacité globale au travers des ordres ponctuels de réalisation interne ou externe.

L'aspect le plus intéressant de ce cadre d'analyse est de mettre en évidence le rôle de la planification à long terme des ressources. On peut en effet concevoir que si une entreprise ne gère pas le niveau stratégique de la décision, elle risque de se trouver acculée de fait dans une situation de choix restreint, soit parce qu'elle aura laissé se détériorer sa capacité interne (pas d'investissement dans les nouvelles technologies, nouveaux développements systématiquement sous-traités), soit au contraire parce qu'elle n'aura pas développé de liens avec des fournisseurs externes ni d'expertise dans la manière de gérer ces liens.

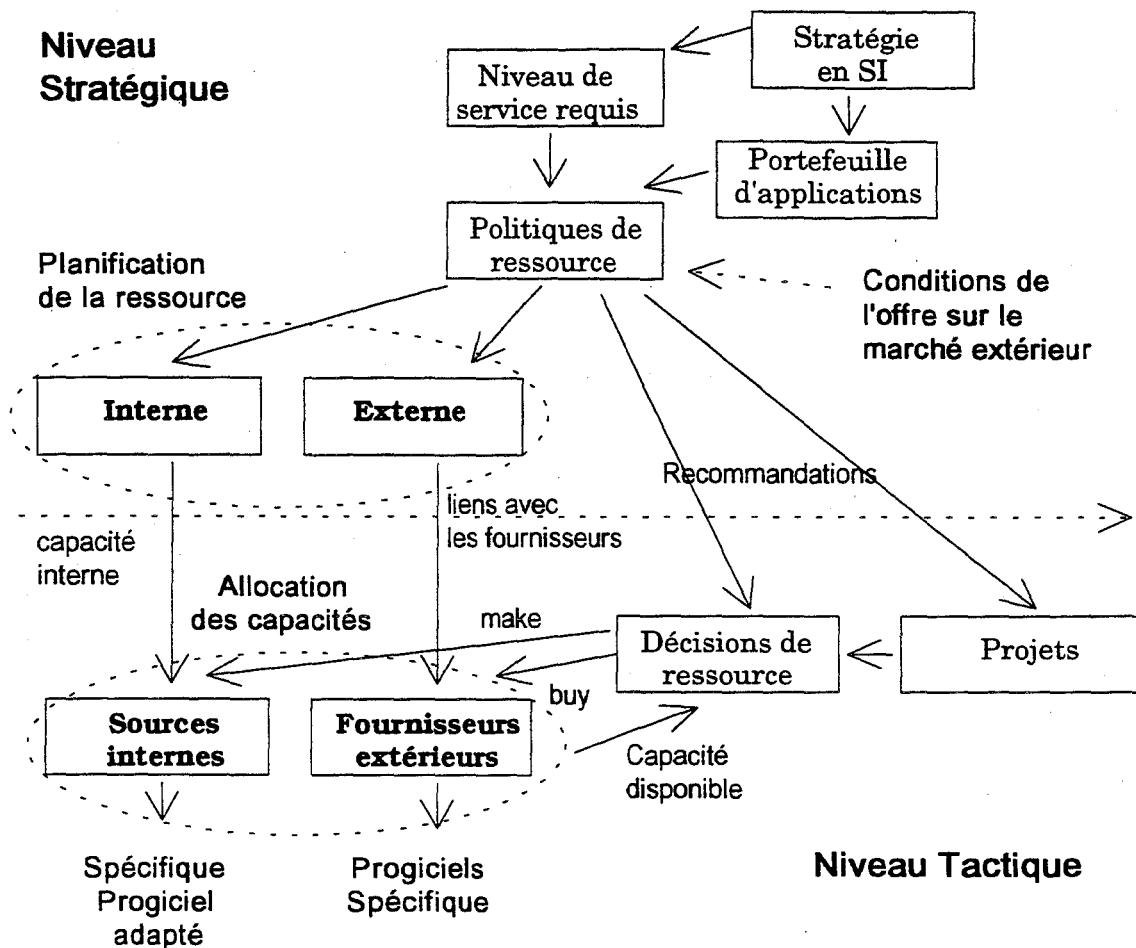


Figure 4.3. Le cadre d'analyse proposé par Rands (1991b) pour les choix de ressources en matière de logiciel

En suggérant de différencier le niveau stratégique (gestion de l'activité) du niveau tactique (gestion des projets), le modèle de Rands permet de clarifier l'analyse des décisions d'externalisation. Il ne répond pas en revanche aux deux questions sous-jacentes que nous allons aborder dans la section suivante :

- comment déterminer, sur un plan stratégique, l'intérêt pour l'entreprise d'internaliser ou d'externaliser l'activité de développement d'applications ?
- les choix d'externalisation opérés au niveau tactique se réduisent-ils à un problème d'allocation de capacité ? Ou, plus précisément, l'entreprise a-t-elle un intérêt stratégique à différencier ses choix d'externalisation selon les projets ?

Section 3 - Le développement d'applications spécifiques présente-t-il un intérêt stratégique fort pour l'organisation ?

Le thème de l'importance stratégique des systèmes d'information a dominé les recherches sur le management des systèmes d'information des dix dernières années. Il a été introduit d'abord par des chercheurs du champ (McFarlan et al., 1983), attesté ensuite par l'un des principaux auteurs en stratégie (Porter et Millar, 1985), analysé par de très nombreux chercheurs (Chokron et Reix, 1987, Ward, 1987, Lederer et Sethi, 1992, Willcocks, 1992), testé empiriquement (avec difficultés) par quelques rares auteurs (Neuman et al., 1992, Francalanci et Maggiolini, 1993), enfin critiqué par d'autres (Hopper, 1990, Huber, 1993).

Nous allons tout d'abord rappeler les grandes lignes de l'analyse de l'impact stratégique des systèmes d'information développée par Porter ; nous nous focaliserons ensuite sur le rôle stratégique des nouveaux développements ; nous terminerons par l'étude du courant qui remet en cause ce rôle stratégique.

I. La reconnaissance de l'impact stratégique des systèmes d'information : la thèse de Porter

L'article de Porter et Millar, "How information gives you competitive advantage" (1985), a été largement repris et commenté ; nous n'en ferons ici qu'un bref rappel. Outre le fait que les tenants de l'impact stratégique des systèmes d'information vont y trouver une source exceptionnelle de légitimité, les travaux de Porter comprennent un ensemble d'apports décisifs :

- Porter détaille tout d'abord la portée stratégique des technologies de l'information tant sur les différentes activités prises individuellement que sur la coordination de l'ensemble des actions. C'est l'analyse de l'**impact sur la chaîne de la valeur** (fig 4.4). En 1985, l'activité de fabrication est un domaine d'application stratégique très important, avec la multiplication des

systèmes XAO². Plus récemment, les **systèmes d'information marketing** ont été également mis en valeur car ils offrent des possibilités de segmentation très fine à partir de bases de données sur la clientèle et constituent donc un très bon support pour des stratégies de différenciation ou de niche (Micheaux, 1992). Un autre résultat intéressant de l'analyse de l'impact sur la chaîne de la valeur est de souligner que la valeur ajoutée peut être accrue par une meilleure coordination ou une accélération des flux (intra- ou inter-entreprises), que permettent les systèmes d'information.

Activités de support	Infrastructure	Modèles de planification, EIS			
	Personnel	Progiciels de gestion du personnel, cartes			
	Technologie	CAO, marché électronique, réalité virtuelle			
	Ressources	Réseaux (marchés mondiaux), EDI			
		Entrepôt automatisé optimisat. transports	CIM, CAO, DAO, FAO, GPAO	Marketing direct Segmentation (BD) Minitel Téléachat	Télé-maintenance, Gestion des pannes, Minitel
	Logistique	Production	Marketing, Vente	Service	
	Activités opérationnelles				Marge

Figure 4.4. L'impact des systèmes d'information sur la chaîne de la valeur
(d'après Porter et Millar, 1985)

- Porter pose également les fondements de l'**analyse sectorielle** de l'impact stratégique des systèmes d'information, à partir de critères originaux et pertinents. Le produit fini est une matrice classant les secteurs sur deux axes : le contenu informatif du produit et l'impact de l'information sur la chaîne de la valeur (figure 4.5). Cette matrice va être très largement utilisée dans les travaux empiriques qui vont suivre, en particulier pour la composition des échantillons

². GPAO (gestion de production), CAO (conception), CFAO (conception et fabrication) assistés par ordinateur forment les systèmes CAD/CAM (computer assisted design / computer assisted manufacturing). On parle maintenant de CIM (computer integrated manufacturing).

car elle permet de traiter la variable "secteur", qui joue un rôle très important dans la plupart des thèmes de recherche en systèmes d'information.

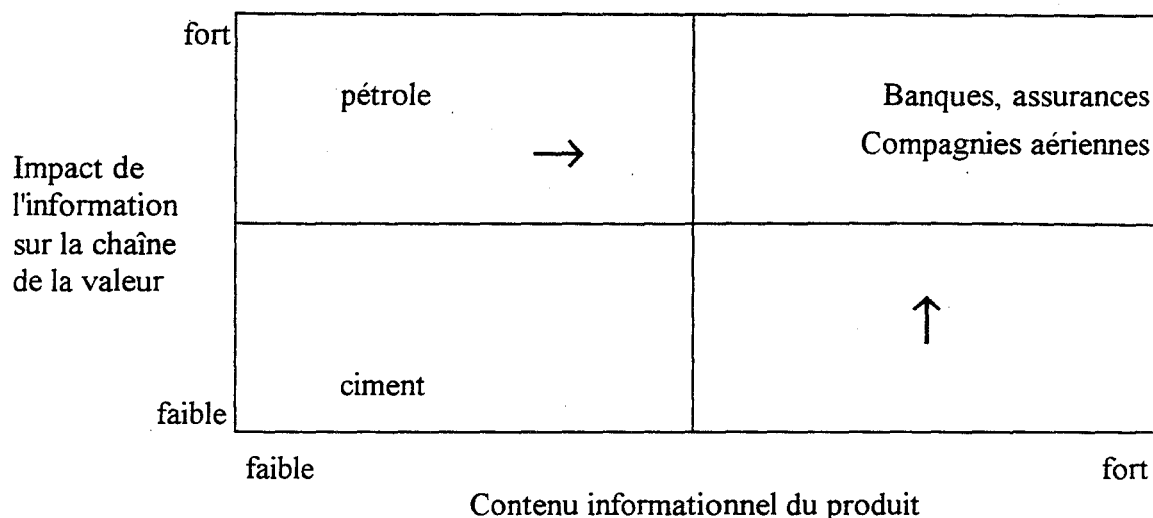


Figure 4.5. L'analyse sectorielle de l'impact des systèmes d'information : la matrice de Porter (source : Porter et Millar, 1985)

Il faut noter que Porter anticipe en 1985 un mouvement de l'ensemble des secteurs vers les cadrans hauts (plus d'information dans la chaîne de la valeur) et droits (plus d'information dans le produit). Les tendances actuelles confirment cette analyse. D'une part, de plus en plus d'activités sont concernées par l'échange de données informatisées, les marchés électroniques ou les systèmes d'information inter-entreprises. D'autre part, les possibilités du multimédia laissent envisager le développement de l'image comme information attachée à une gamme de plus en plus étendue de produits.

- enfin, il est affirmé que le choix d'investissements informatiques judicieux peut être la source d'un **avantage concurrentiel durable**. Dans son argumentation, Porter souligne que les principaux avantages vont aux premiers acteurs capables d'exploiter de nouvelles opportunités; les changements induits par de nouveaux systèmes d'information peuvent conduire à une redéfinition de la structure de la concurrence dans le secteur. L'essentiel est

alors de devancer les concurrents en anticipant et en initiant les changements dus à l'informatique.

II. Le rôle des nouveaux développements

Pour McFarlan, McKenney et Pyburn (1983) il est nécessaire de faire la distinction entre l'importance stratégique **actuelle** des systèmes d'informations pour une organisation et l'importance stratégique **future** :

- l'importance stratégique actuelle correspondant principalement à l'estimation du poids de l'informatique dans les opérations courantes - applications existantes et fonctions d'exploitation,

- l'importance stratégique future peut être estimée à partir des **nouveaux projets de développements applicatifs**.

Chaque entreprise peut ainsi être classée sur une grille à quatre cadrans exprimant la position stratégique de son système d'information (figure 4.6).

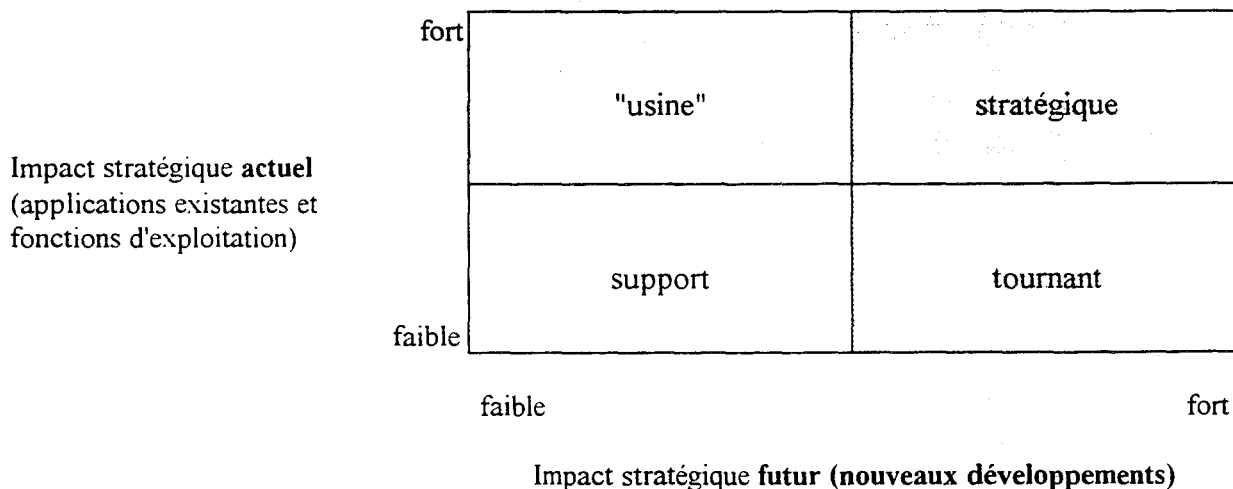


Figure 4.6. L'importance stratégique actuelle et future des systèmes d'information
(D'après McFarlan et al., 1983)

Le rôle stratégique des nouveaux développements est limité dans deux des cas identifiés :

- l'impact actuel et futur faible réduit les systèmes d'information à un rôle de support. Ils ne font dans ce cas pas partie du coeur de l'activité de l'organisation.

- un impact actuel fort mais peu susceptible d'offrir des opportunités significatives dans l'avenir correspond à la position "usine" : le système d'information joue un rôle essentiel dans les activités quotidiennes mais ce rôle est peu susceptible d'évoluer vers la définition de nouvelles activités ou la création de nouveaux avantages concurrentiels.

En revanche, les nouveaux développements sont stratégiques dans les deux autres cas :

- un impact actuel faible mais susceptible de se développer fortement dans l'avenir correspond à la situation de "tournant". Le développement des nouvelles applications devient vital pour l'entreprise,

- un impact actuel et futur fort correspond à une position stratégique des systèmes d'information, qui peuvent être considérés comme une ressource de base de l'organisation.

Les auteurs placent le secteur d'activité parmi les principaux facteurs pouvant influencer la situation d'une firme sur la grille : des secteurs "à base d'informations" comme la banque ou l'assurance sont extrêmement dépendants des fonctions quotidiennes d'exploitation informatique et ont des projets de développement vitaux pour leur capacité concurrentielle future ; dans d'autres secteurs l'informatique est réduite à un rôle de support. Ils rejoignent en cela l'analyse de Porter et Millar.

Cependant leur analyse prend en compte d'autres facteurs :

- la présence ou non de **nouvelles opportunités** dans le secteur considéré ; certains secteurs peuvent dépendre fortement des systèmes d'information mais avoir atteint un stade de maturité où les nouvelles applications sont peu susceptibles d'accroître la rentabilité ; les auteurs donnent l'exemple des compagnies aériennes,

- le stade de développement atteint par la firme : le système d'information joue un rôle plus important en phase de croissance (**portefeuille applicatif en développement**) que de maturité,

- le type de stratégie adoptée par l'entreprise (stratégie de coût, de différenciation ou recherche de niches) : **le système d'information ne sera stratégique que s'il facilite la mise en oeuvre de la ligne stratégique choisie.**

Ward (1987) a prolongé les travaux de McFarlan, McKenney et Pyburn en proposant une "gestion stratégique du portefeuille d'applications". L'auteur propose d'analyser le rôle stratégique de **chaque** application existante et de chaque projet de développement et de le positionner sur la grille en fonction de l'impact stratégique actuel et futur.

Il obtient ainsi une classification³ des applications de l'entreprise dans les quatre groupes stratégiques en fonction des bénéfices générés et des ressources consommées (fig 4.7).

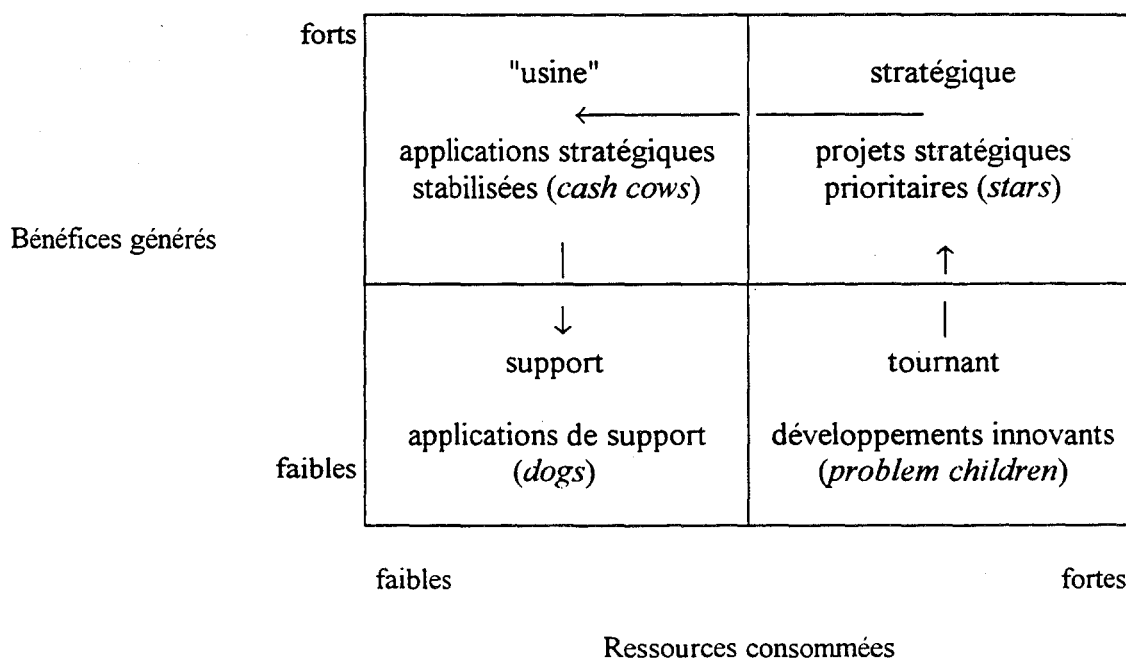


Figure 4.7. Analyse stratégique du portefeuille d'applications (source Ward, 1987)

- les **projets stratégiques prioritaires (*stars*)** génèrent des profits pour l'organisation mais requièrent un niveau d'investissement élevé ; ils sont situés dans le cadran stratégique (impact actuel et futur fort),

³. L'auteur reprend la terminologie classique des matrices BCG.

- les "**applications stratégiques stabilisées**" (*cash cows*) sont des applications génératrices de profits mais qui sont arrivées à maturité ; la consommation de ressources de développement est faible,

- les **applications de support** (*dogs*) sont obligatoires mais doivent consommer le moins de ressources possibles,

- les **développements innovants** (*problem children*) requièrent des investissements élevés financés par les économies réalisées sur les applications de support.

Au fur et à mesure qu'un type d'application se banalise, il évolue dans les 4 cadrans : l'auteur donne l'exemple des systèmes de gestion automatisés de la paie qui ont été les applications innovantes des années 60, ont eu ensuite un rôle stratégique en donnant une image de modernité aux premières compagnies qui les ont mis en oeuvre, ont permis pendant des années d'améliorer les procédures des services concernés et de réaliser des économies de gestion et sont finalement passées dans la plupart des cas à un rôle de support nécessaire mais sans nouvelle valeur ajoutée. Les applications de support sont largement banalisées ; elles sont maintenant généralement prises en charge par des progiciels.

L'intérêt de la classification de Ward est de permettre une gestion différenciée de l'activité de développement directement appuyée sur une vision stratégique. Cela revient à assigner des objectifs différents en fonction des applications et projets :

- pour les applications stratégiques stabilisées on recherchera avant tout la **rentabilité** des nouveaux développements,

- pour les applications de support on s'orientera vers une **limitation des coûts**,

- pour les projets stratégiques prioritaires, l'objectif est de développer les compétences internes et de rechercher systématiquement les axes d'amélioration permettant de créer un avantage concurrentiel - c'est une **compétence de base** pour l'organisation,

- le maintien en interne d'une capacité à initier des projets innovants est l'**investissement nécessaire pour entretenir la compétence de base** de l'entreprise.

Cela revient aussi à avoir une **politique d'externalisation différenciée selon les projets de développement** :

- privilégier la ressource interne pour les développements stratégiques et les développements innovants à fort potentiel,
- utiliser l'externalisation sur les autres projets à la fois pour libérer les ressources (hommes, compétences) sur les projets prioritaires et pour dégager des économies afin de financer les investissements sur les projets innovants et le maintien des connaissances.

Cependant, la thèse de l'intérêt stratégique du développement d'applications en particulier n'a pas que des défenseurs ; nous allons maintenant étudier les positions de ses détracteurs.

III. La remise en cause de l'intérêt stratégique

Malgré la solidité des concepts et des outils développés, les thèses que nous venons d'étudier ne sont pas à l'abri de critiques souvent pertinentes. Le premier reproche concerne le manque de validation empirique, dû aux difficultés rencontrées dans la définition de variables opérationnelles permettant de **mesurer** l'impact stratégique. La deuxième série de reproches remet plus fondamentalement en cause la **possibilité de créer un avantage concurrentiel** durable appuyé sur des développements spécifiques ou les moyens d'y parvenir.

3.1 les difficultés de mesure de l'impact stratégique

Le problème de mesure de l'impact stratégique des systèmes d'information rejoint celui, plus général, de l'évaluation des performances et de la rentabilité des investissements en systèmes d'information. Malgré les efforts entrepris par de nombreux chercheurs et praticiens, les difficultés demeurent :

- on manque tout d'abord de variables objectives pour opérationnaliser ces concepts. La plupart des mesures de performance du système d'information utilisées dans les recherches empiriques sont indirectes et perceptuelles (on demande au manager ce qu'il pense de l'impact stratégique ou encore quel est son degré de satisfaction par rapport au système). Certaines

recherches ont cependant conduit à la définition d'outils opérationnels intéressants, comme les grilles d'évaluation des systèmes d'information établies par C. Dumoulin (1986).

- les tentatives de quantification directe ne donnent que des résultats limités.

On peut distinguer ici deux approches. Une première direction, adoptée principalement par les praticiens d'entreprise, consiste à tenter d'évaluer la rentabilité directe des projets informatiques avant de prendre la décision d'engagement. Mais dans la pratique, seules les baisses de coûts directement induites et quantifiables⁴ sont prises en compte dans l'évaluation *a priori* des bénéfices. De plus, la rentabilité des projets est rarement contrôlée *a posteriori*.

Une deuxième approche, plus cohérente avec l'analyse de l'impact du système d'information sur la chaîne de la valeur développée par Porter, cherche à mesurer cet impact à un niveau très global, pour prendre en compte les effets indirects. Strassman (1990) a ainsi développé le concept de *ROM (return on management)*. Il part du principe que le bénéfice majeur induit par un nouveau système d'information est d'améliorer la performance des managers (par une meilleure information, une communication plus rapide et mieux ciblée, de nouveaux choix d'organisation...). Il propose donc une mesure globale de la valeur ajoutée du management pour la firme. La difficulté est alors d'établir un lien direct entre l'amélioration de la performance (dont on peut malgré tout penser qu'elle peut être due à d'autres facteurs) et les investissements dans le système d'information. La validation proposée est statistique ; elle n'a pu être obtenue pour l'instant que dans certains cas limités même si certaines recherches paraissent prometteuses (Francalanci et Maggiolini, 1993). En particulier, il semble nécessaire que les entreprises aient une gestion déjà efficace pour bénéficier des avantages liés aux investissements en systèmes d'information.

De plus, il faut noter que la globalité de cette mesure ne la rend pas très attractive pour des praticiens confrontés aux décisions de gestion des systèmes d'information, qui l'utilisent rarement (Willcocks, 1992).

⁴. C'est-à-dire concrètement le nombre de postes de travail qui pourront être supprimés, ce qui n'améliore pas l'image de l'informatique en temps de crise et constitue un argument de vente des nouveaux projets de plus en plus difficile à défendre.

Cette insuffisance de preuves tangibles de la performance des systèmes d'information et de mesures reconnues de leur impact stratégique est d'autant plus dommageable que les coûts des systèmes ont tendance à s'accroître⁵ et se mesurent en pourcentage de chiffre d'affaires, unité bien concrète, elle, aux yeux des managers, en particulier en temps de crise.

3.2 La remise en cause du principe de l'acquisition d'un avantage concurrentiel durable par le biais du système d'information

Le manque de validation empirique n'est pas le seul reproche adressé aux modèles stratégiques. Certains auteurs ont remis en cause le principe même de l'acquisition d'un avantage concurrentiel durable par le biais des systèmes d'information. Les arguments évoqués à l'appui de cette thèse sont issus de plusieurs écoles de pensée, qui s'appuient sur des logiques assez différentes.

Cole (1984) reconsidère le lien entre le **développement de nouveaux systèmes d'information** et l'amélioration des résultats. Il s'inscrit dans une lignée de chercheurs qui, au-delà des systèmes techniques, soulignent le poids des conditions organisationnelles et les limites cognitives individuelles⁶. Pour Cole, l'impact stratégique espéré est trop souvent le prétexte à l'élaboration de systèmes inefficaces parce que trop complexes et mal intégrés à l'organisation. Il y a risque de surenchère technologique et de sur-information. De plus, l'amélioration de la communication souhaitée n'est pas obtenue quand les flux d'informations restent hiérarchiques et non transversaux et quand l'information est collectée davantage dans un souci d'évaluation que de décentralisation des responsabilités. Cole préconise donc des systèmes d'information simples, proches des niveaux de décision et pas forcément informatisés,

⁵. Encore ne parle-t-on ici que des coûts directs (matériel, logiciel, mod, formation). Le processus de mise en oeuvre de nouvelles applications comporte également des coûts indirects (gestion du changement) qui ne sont souvent pas pris en compte mais qui renforcent les questions sur la rentabilité de ces opérations.

⁶. Le célèbre article de Ackoff. "Management MISinformation Systems" (1967) est bien représentatif de ce dernier point.

à l'exemple de ceux qu'il croit discerner dans les entreprises japonaises. Nous aurons l'occasion de revenir sur certains aspects des critiques de l'auteur, en particulier les limites cognitives individuelles. Pour l'instant, soulignons que les critiques adressées par Cole concernent davantage l'accompagnement humain et organisationnel des projets et la culture managériale américaine que l'impact stratégique des systèmes d'information. Il reconnaît par ailleurs l'importance stratégique de systèmes d'analyse des marchés ou d'étude du comportement des consommateurs. Certains systèmes d'information seraient donc stratégiques, le problème étant de bien identifier lesquels et d'accompagner leur mise en oeuvre par une structure organisationnelle favorisant la prise de décision au niveau opérationnel.

L'argument de Hopper (1990) est assez différent. Pour lui, à peine débutée, l'ère des systèmes d'informations stratégiques pour l'entreprise est déjà en voie d'achèvement. Pour comprendre son raisonnement, il faut rappeler que Hopper exerce son activité dans le secteur de l'aéronautique⁷. Les systèmes de réservation automatisés ont transformé le marketing et la vente de ce secteur qui a été parmi les tout premiers à se restructurer profondément du fait de la modification des systèmes d'information. Au moment de cette restructuration, *American Airlines* a obtenu un avantage concurrentiel considérable grâce au système SABRE, développé en interne. Cependant, depuis 1986, l'entreprise vend à ses concurrents son système de *yield management* (optimisation des capacités et des tarifs). SABRE est devenu l'un des premiers supermarchés électroniques, spécialisé dans le voyage, utilisé par de multiples compagnies et qui, aux dires de Hopper, vend maintenant les mêmes fonctionnalités à tous et n'avantage pas spécifiquement son propriétaire.

Plusieurs leçons peuvent être tirées de cette expérience :

⁷. Au moment où il écrit cet article, Hopper est *senior vice president* en charge des systèmes d'informations chez *American Airlines*.

- Tout d'abord, American Airlines a effectivement pu bénéficier d'un avantage car elle a été la **première** à proposer un système d'information optimisé. Il est clair que ce type d'avantage n'est accessible qu'aux deux ou trois concurrents d'un secteur qui saisissent les premiers ce type d'opportunité. Pour tous les autres, le risque serait plutôt d'engager des fonds importants dans la mise en oeuvre d'un système propriétaire ; la prudence conseille alors d'adopter au contraire une position d'attente, le temps que la restructuration soit suffisamment engagée pour savoir quel système extérieur rejoindre.

Il faut souligner que l'aéronautique, secteur dans lequel Hopper exerce son activité professionnelle, est l'un des rares à avoir atteint une certaine maturité quant au développement et à l'usage des nouvelles technologies de l'information. Les nouvelles cartes sont donc distribuées et la compétition s'est reportée sur d'autres aspects, sur lesquels les systèmes d'information ont peu d'impacts. L'idée d'une "maturité" d'un secteur - au moins temporaire, avant l'apparition de nouvelles opportunités - est intéressante et suggère qu'il y a des moments davantage propices que d'autres à l'investissement dans de nouvelles applications⁸. On retrouve ici l'analyse stratégique du portefeuille d'applications de Ward : si un secteur est pour un moment en phase de stabilité après une période très active de redéfinition de la concurrence appuyée sur les systèmes d'information, il y a de fortes chances d'observer une "pause" dans les investissements en nouveaux développements, la priorité étant à la rentabilisation des applications stratégiques stabilisées.

- Selon Hopper, l'expérience ne serait de toute façon plus reproductible actuellement, les nouveaux systèmes d'information exigeant un apport de ressources hors de portée d'une seule entreprise. Les nouveaux systèmes stratégiques deviendraient donc, pour de simples raisons d'économies d'échelle, le **fruit de coopérations** entre plusieurs firmes concurrentes au sein

⁸. Hopper va même jusqu'à évoquer la "maturité ultime" qui ne serait autre que la banalisation et dont il dresse un tableau volontairement provocateur : *"Eventually -and we are far from this time- information systems will be thought of more like electricity or the telephone network than as a decisive source of competitive advantage. In this world, a company trumpeting the appointment of a new chief information officer will seem as anachronistic as a company today naming a new vice president for water and gas"*.

d'un même secteur, ce qui change la façon d'envisager leur impact et surtout remet en cause le lien avec le portefeuille de développement individuel de la firme établi par McFarlan.

- Hopper évoque également l'argument selon lequel l'important n'est pas de construire son propre système mais plutôt de sélectionner le meilleur système d'information disponible sur le marché et d'exceller dans son **utilisation**. L'analyse sous-jacente est que l'expertise utile n'est plus celle du *développement* de systèmes mais celle des *applications* de systèmes standard à l'activité de l'entreprise.

Dans le même sens, Huber, vice-président et administrateur de la Continental Bank, explique dans un article récent (1993) pourquoi la banque a choisi d'externaliser les "joyaux de la couronne" (c'est-à-dire son informatique). La logique de son argumentation tient en deux propositions : (1) les systèmes informatiques présentent un **intérêt stratégique fort** pour l'organisation, et (2) le fait de gérer cette activité en interne **ne permet pas de générer de façon optimale des avantages concurrentiels**.

Huber reprend l'argument de la *qualité* de la prestation extérieure, supérieure à la prestation interne en termes de flexibilité, de rapidité et de connaissance des technologies. De plus, il dissocie la fabrication en interne du système d'information de l'aptitude à utiliser le système d'information à des fins stratégiques, ce qu'il exprime sous une forme imagée : "N'y a-t-il pas de bonnes raisons à ce qu'un pilote de Formule 1 conduise son véhicule au lieu de le construire ?" (Huber, 1993). L'auteur évoque en outre la difficulté de protéger l'investissement de développement alors que les logiciels applicatifs peuvent être "clonés" en quelques semaines.

Peut-on développer une expertise d'utilisation d'applications informatiques sans s'appuyer sur une expertise de fabrication ? Dans le cas d'*American Airlines*, on peut raisonnablement penser que l'apprentissage organisationnel s'est effectué pendant la phase de conception et de réalisation du produit et porte ses fruits en phase d'utilisation. En d'autres termes, c'est *parce que American Airlines* a consacré de fortes ressources internes à la fabrication de SABRE que Hopper peut dire qu'elle en est la "meilleure utilisatrice au monde". Mais est-ce vrai d'un

logiciel de trésorerie, d'un système de gestion de production, d'une application marketing ? Où se situe la connaissance utile ?

Pour notre part, nous voyons plusieurs limites à ce raisonnement :

- les applications disponibles sur le marché sont en règle générale des applications déjà banalisées, peu susceptibles de générer un nouvel avantage concurrentiel (on retrouve ici les limites des progiciels).

- adopter une optique d'utilisation et d'actualisation systématique risque à terme de faire perdre à l'entreprise sa capacité à anticiper la relation entre son marché et les opportunités offertes par les nouveaux développements. Si l'on reprend l'analogie de Huber, force est de constater que les pilotes de formule 1 ne se contentent pas de conduire leurs voitures mais jouent un rôle majeur dans leurs évolutions techniques car ils sont à la meilleure place pour faire l'interface entre l'évolution du marché (configuration des circuits, capacités des concurrents) et les spécifications du produit. On revient ici au concept de "compétence architecturale" développé par Venkatesan.

Conclusion

Le tableau 4.1 fait la synthèse des différents éléments abordés dans ce chapitre.

Nous rappellerons ici les principaux éléments :

L'approche stratégique de la ressource nous semble **adaptée** au problème particulier de l'activité de développement d'applications.

Dans ce cadre, le choix entre ressource interne ou externalisation est lié à **l'intérêt stratégique** des activités de développement d'applications pour l'organisation : on privilégiera le maintien en interne de la partie identifiée comme une des "ressources de base" stratégique de l'organisation et l'externalisation du reste de l'activité.

Cet intérêt stratégique dépend de différents critères ; pour notre part, nous retenons trois éléments principaux :

- l'intérêt stratégique du développement d'applications est variable selon les **secteurs d'activité**,

- au sein du portefeuille d'applications, l'intérêt stratégique est variable selon **les projets**,

- au sein du processus de développement, l'intérêt stratégique est variable selon les **étapes**.

Il est élevé pour l'étape d'analyse du besoin, dans laquelle s'exerce la compétence architecturale et qui peut être considérée comme une phase critique du processus de développement. Il est plus faible pour les étapes d'exécution qui mettent en oeuvre un savoir technique.

Principaux éléments de l'approche stratégique	Application à l'activité de développement d'applications	Difficultés et limites
<p>Gestion stratégique de la ressource</p> <p>1. Identifier les "ressources de base" formant le coeur du métier et les développer en interne.</p> <p>2. Privilégier l'externalisation des ressources secondaires pour permettre une meilleure focalisation vers les objectifs stratégiques.</p> <p>3. S'assurer de la disponibilité et de la maîtrise des ressources externalisées.</p>	<p>Le développement d'applications fait-il partie des ressources de base de l'organisation ? (Question de l'intérêt stratégique).</p> <p>L'intérêt stratégique est variable :</p> <ul style="list-style-type: none"> - selon les secteurs d'activité (Porter), - selon les projets au sein du portefeuille de développement (Ward) <ul style="list-style-type: none"> . les développements stratégiques prioritaires et les nouveaux développements importants font partie des ressources de base --> privilégier les solutions internes . les applications de support et certaines applications stabilisées peuvent être considérées comme des ressources secondaires --> externalisation à envisager - selon les étapes du processus (Venkatesan). 	<p>Remise en cause de l'intérêt stratégique du développement d'applications :</p> <ul style="list-style-type: none"> - difficulté de mesure de l'impact, - avantage concurrentiel limité aux innovateurs, - possibilité d'acheter le logiciel et de valoriser son utilisation. <p>Difficulté d'évaluation des projets stratégiques.</p> <p>Risque de défaillance du fournisseur, de perte de contrôle, d'augmentation des coûts à long terme.</p>

Conditions de mise en oeuvre de l'externalisation	Variables à prendre en compte dans la décision d'externalisation
<p>Politique de ressource (Rands).</p> <p>Maintien en interne de la "compétence architecturale" (Venkatesan)</p> <ul style="list-style-type: none"> - capacité à traduire les besoins en langage de spécification technique, - connaissance approfondie de l'impact de la technologie sur les marchés. 	<ul style="list-style-type: none"> • secteur d'activité • importance stratégique du projet • étape du processus.

Tableau 4.1 Synthèse de l'approche stratégique

Chapitre 5 :
L'étude de la relation contractuelle -
les apports de la théorie de l'agence

Introduction

Nous avons abordé pour l'instant le problème de l'externalisation du développement d'applications sous l'angle industriel et stratégique. Mais l'externalisation suppose également la gestion d'une relation contractuelle, qui peut poser en elle-même des problèmes à l'organisation. Cette perspective contractuelle a donné lieu depuis quelques années à un nouvel ensemble particulièrement riche de travaux : le point central de ces recherches est **l'étude de la relation d'échange et la gestion de la transaction**. L'idée principale est que l'organisation encourt pour assurer l'ensemble de ses activités non seulement des coûts de production (pris en compte dans l'approche industrielle classique) mais également des coûts de gestion des échanges, qui doivent et peuvent être optimisés par le choix d'une solution contractuelle adaptée.

La richesse des travaux de ce champ d'étude nous conduit à consacrer deux chapitres à la perspective contractuelle.

L'objectif de ce premier chapitre est de **définir et de modéliser la relation contractuelle**. Nous adopterons pour cela une démarche en deux temps :

- dans un premier temps, nous chercherons à établir les **caractéristiques juridiques de l'objet de la transaction, le logiciel spécifique, et des transactions afférentes**. Quelles sont les caractéristiques contractuelles du développement d'applications ? Comment le logiciel est-il défini par le droit ? Quels sont les droits de propriété ? Quelles sont les règles qui gouvernent les contrats en matière de logiciel ? Nous verrons sur ce dernier point que s'est développé autour du logiciel depuis une vingtaine d'années un contexte particulier d'application du droit au travers d'une jurisprudence très riche. En conclusion de cette étude juridique nous recenserons un ensemble de caractéristiques du produit (le logiciel applicatif) et des transactions.

- dans un second temps, nous examinerons les **apports de la théorie de l'agence** pour notre travail. L'intérêt de la théorie de l'agence est de présenter une modélisation très structurée de la relation contractuelle ; elle nous permettra de mettre en évidence certains risques comportementaux et un ensemble de coûts liés à la situation de délégation d'une tâche. Nous conclurons sur la théorie de l'agence en analysant les deux formes contractuelles types qu'elle propose : contrats basés sur l'évaluation du comportement (*behavior-based contracts*) et contrats basés sur l'évaluation du résultat (*outcome-based contracts*), qui correspondent de façon quasi parfaite aux contrats de sous-traitance informatique en régie et au forfait.

Le chapitre 6, qui forme le deuxième volet de notre analyse contractuelle, analysera quant à lui les apports pour notre travail de la théorie des coûts de transaction développée par O. Williamson.

Section 1 - Le logiciel, objet contractuel

L'élaboration d'une définition précise du logiciel a été un enjeu important des vingt dernières années pour les juristes, l'objectif étant de classer le logiciel dans les diverses catégories du droit, ce afin de répondre aux évolutions des pratiques des agents économiques : il a ainsi fallu examiner la question des droits de propriété ou encore définir les obligations contractuelles attachées à la fourniture de logiciels. Ces réflexions se sont concrétisées dans un ensemble de textes précis et d'arrêtés de la jurisprudence, dont nous reprenons ici les principaux éléments. Précisons tout de suite que notre objet n'est pas de chercher à explorer en détail les arcanes du raisonnement juridique. Nous ne retiendrons donc ici que les éléments qui peuvent nous éclairer sur la nature juridique du logiciel et les contrats dont il fait l'objet. Nous aborderons successivement ces deux points, en nous référant principalement aux travaux de Toubol (1986) et de Lamberterie (1983) ainsi qu'au LAMY droit de l'informatique (1992).

I. L'identification du logiciel comme objet du droit

Dans son ouvrage consacré à l'analyse juridique du logiciel, Toubol (1986) retrace les étapes du difficile cheminement du droit dans la recherche d'une qualification appropriée pour le logiciel. Différentes "catégories" juridiques ont ainsi été examinées pour savoir si le logiciel était susceptible de s'y intégrer, la réponse à cette question divergeant d'ailleurs selon les pays. Si la qualification retenue présente un intérêt évident pour l'analyse des transactions, l'examen des arguments qui ont conduit à retenir ou à écarter telle ou telle qualification nous renseigne également sur la nature du logiciel. Selon Toubol, trois thèmes ont dominé le débat sur la qualification juridique du logiciel : la classification en tant qu'objet immatériel, l'assimilation à une invention industrielle et la protection par droits d'auteurs.

1.1 Le logiciel est un objet immatériel

Il s'agit d'une "valeur économique qui n'est pas composée de matières et n'est pas préhensible mais qui se caractérise par le fait qu'elle s'adresse à l'esprit". Cette analyse permet de classer le logiciel parmi les biens incorporels, caractérisés par leur objet immatériel et par le fait qu'ils résultent de l'activité intellectuelle des hommes. Plus précisément, Toubol qualifie le logiciel "d'outil immatériel" : le logiciel est un objet fabriqué qui sert à réaliser un travail donné (vocation utilitaire) tout en étant par essence composé d'informations, dont la substance est incorporelle. Il est donc à la fois concret et abstrait. Cette dualité troublante est renforcée par la difficulté de perception par l'esprit humain d'une chose qui s'adresse à une machine : la médiatisation obligatoire par la machine -système concret mais dont la technique est obscure pour les non-spécialistes- renforce paradoxalement la perception d'immatérialité et d'abstraction.

1.2 La question de l'assimilation à une invention industrielle

Le logiciel a été écarté très tôt en France -dès 1968- de la protection par brevet à laquelle ont droit les inventions industrielles¹. Il apparaît que le législateur a éprouvé des difficultés tant pour apprécier la **nouveauté** et **l'activité inventive** (caractéristiques qui déterminent le statut d'invention) que pour attester le **caractère industriel** du logiciel :

- la difficulté d'appréciation de la nouveauté (critère objectif) ressort principalement d'un problème de compétence. Les législateurs ont ainsi estimé qu'il serait techniquement difficile de juger de la nouveauté d'un programme. De plus, la technologie du logiciel est en continuelle évolution ce qui fait que de nombreuses équipes travaillent sur les mêmes sujets ; il n'est pas forcément souhaitable que toute nouveauté, même marginale, soit brevetée.

¹. Cette décision pourrait être remise en question pour des raisons économiques : la jurisprudence américaine admet en effet la protection du logiciel par *copyright* et par brevet et des milliers de dépôts de brevets portant sur des logiciels sont affectés chaque mois aux Etats-Unis, ce qui peut désavantager les industries des pays n'offrant pas le même type de protection (Expertises des systèmes d'information, Avril 1993).

- la question de l'activité inventive (critère subjectif) est également embarrassante. Toubol, passant en revue les étapes de la conception d'un logiciel, a cherché à déterminer à quel moment pourrait intervenir une activité inventive : "Recenser les fonctionnalités auxquelles le programme devra répondre nécessite un travail méticuleux et difficile mais ne révèle aucunement une activité inventive. L'élaboration de l'analyse organique qui consiste à trouver les moyens qui permettront de répondre aux besoins recensés n'est pas toujours évidente par rapport à l'état de la technique et est donc susceptible de révéler une activité inventive. La traduction de l'organigramme en langage informatique ou la programmation peut être effectuée de façon plus ou moins astucieuse mais ne suppose généralement pas d'activité inventive.. l'évolution permet de penser que l'activité de programmation sera de plus en plus mécanique.. Ce n'est qu'au niveau de l'organigramme qui se présente comme un procédé, que l'activité inventive de l'informaticien peut se développer et qu'il peut donc y avoir une invention" (pp.44-45).

L'auteur met ainsi en avant la possibilité d'une forme de "créativité technique" qui serait celle de l'ingénieur qui choisit des moyens et élabore la structure d'une réalisation. Cependant la référence à l'organigramme paraît discutable ; il nous semble que l'activité inventive peut également s'exercer dans l'identification de nouvelles fonctionnalités ou la définition de l'interface utilisateur.

Nous retrouvons ici certains éléments déjà évoqués dans le chapitre 3 à propos du processus de développement : dans quelle mesure la production d'un logiciel relève-t-elle de l'application habile de règles et de la mise en oeuvre d'un savoir-faire ou au contraire d'un processus de création ? Le législateur reconnaît le statut de technique très complexe au logiciel, au point de l'exclure du domaine des brevets pourtant habitué à traiter de la complexité technique. Cependant, tout comme les acteurs en entreprises, il hésite à faire la part entre ce qui ressort d'un processus compliqué mais connu, voire mécanique, et ce qui serait du domaine de l'action créatrice.

- le caractère industriel enfin a été refusé au logiciel par le législateur. Il semble que c'est en raison de la "trop grande abstraction des résultats produits par les programmes" (Toubol, p.53) que ceux-ci ont été au départ écartés, tant aux Etats-Unis qu'en France, de la brevetabilité. C'est l'**immatérialité des résultats** produits par un programme d'ordinateur qui est ici mise en évidence, même quand celui-ci est un outil pouvant permettre d'améliorer un processus industriel². Le caractère abstrait de l'objet, la difficulté d'évaluer les résultats sont ainsi à la base du traitement juridique du logiciel.

1.3 La protection par droits d'auteur

Les droits d'auteur protègent les "oeuvres de l'esprit, quels qu'en soient le genre, la forme d'expression, le mérite ou la destination" (loi du 11 mars 1957 sur la propriété littéraire et artistique). Depuis la loi du 3 juillet 1985, le logiciel a été ajouté à la liste des oeuvres protégées. Pour avoir droit à la protection par les droits d'auteur, il faut pouvoir attribuer au logiciel la potentialité **d'originalité**. Cette potentialité a été reconnue et attestée dans un arrêté de la jurisprudence devenu célèbre où elle est interprétée comme "la marque de l'apport intellectuel" de l'auteur du logiciel ; ce dernier "avait fait preuve d'un effort personnalisé allant au-delà de la simple mise en oeuvre d'une logique automatique et contraignante et la matérialisation de cet effort résidait dans une structure individualisée" (arrêt de la Cour de cassation du 7 mars 1986, affaire Babolat c. Pachot).

Il reste néanmoins que l'assimilation du logiciel à l'oeuvre -"la forme sensible sous laquelle toute création se manifeste et devient accessible à autrui"- est surprenante. Toubol souligne que le logiciel est, par essence, plus proche d'une invention brevetable que d'une oeuvre de l'esprit (p.56). Aussi faut-il interpréter cette décision en invoquant les motifs économiques de ce choix qui, dès lors que la protection par brevet était rendue difficile,

². Il faut cependant noter que le régime des brevets a été assoupli depuis dans le cadre de l'office Européen des brevets et permet d'inclure un programme d'ordinateur dans un brevet, dès lors que l'effet du programme est de caractère technique.

permet de bénéficier des conventions internationales de Berne et de Genève qui protègent les oeuvres de l'esprit. Cependant les logiciels sont des créations techniques ; cette particularité, ainsi que le souci de prendre en compte les spécificités économiques ont abouti à ce que Toubol appelle "un régime des droits d'auteur défiguré", c'est-à-dire dérogeant au droit de suite du créateur sur l'oeuvre créée et inversant le principe de domination de l'exécutant sur l'instigateur : les salariés développant un logiciel dans l'exercice de leurs fonctions sont ainsi exclus du bénéfice du droit d'auteur au profit de leur employeur.

Cette situation aboutit à un régime de droits de propriété complexe, susceptible d'évolution parce qu'il est non homogène entre les différents pays qui produisent et utilisent des logiciels, et enfin comportant des zones d'incertitude que les parties doivent préciser contractuellement : par exemple, la loi française ne traite pas explicitement le cas du logiciel spécifique effectué sur commande par un fournisseur pour son client, dans le cadre d'un contrat d'entreprise. La jurisprudence attribue de fait en l'absence d'une clause contractuelle expresse la propriété de l'oeuvre à son créateur (TGI Versailles, affaire Harmand c. Guedy, 19-11-1991). *Une entreprise, sauf clause contractuelle expresse, ne possède qu'un droit d'usage sur le logiciel qu'elle a fait réaliser à l'extérieur et ne peut s'opposer à sa diffusion, y compris chez ses concurrents.*

II. Les dispositions du droit des contrats en matière de logiciel

Les dispositions du droit des contrats en matière de logiciel nous intéressent doublement. Tout d'abord, parce qu'elles sont susceptibles d'éclairer notre définition du logiciel, et ensuite parce que ces dispositions interviennent comme "toile de fond" dans l'organisation des échanges entre firmes. Cet examen est d'autant plus nécessaire que, alors qu'il n'existe pas de règles spécifiques aux contrats informatiques, il s'est développé à travers une jurisprudence abondante un *contexte particulier d'application du droit*, formé par un ensemble original d'obligations pour le client et le fournisseur.

Avant d'entrer dans le détail de ces spécificités, il faut noter qu'avec le droit des contrats se manifeste la nécessité de distinguer différentes catégories de logiciel et différentes catégories de prestations (d'échange) : il y a peu de choses communes en effet entre la *shrink-wrap license* (également connue sous le nom de *tear-me-open*) développée aux Etats-Unis pour certains progiciels selon laquelle défaire l'emballage implique l'acceptation des termes de la convention (Lamy, n°318) et le contrat qui porte sur le développement d'un logiciel spécifique, qui seul nous intéresse ici.

2.1 Les différentes catégories de logiciel

Le Lamy du droit de l'informatique distingue deux catégories de logiciel, les logiciels "déjà élaborés" et les logiciels "à élaborer". Pour les logiciels applicatifs, ceci correspond à l'opposition classique entre "progiciel" et "spécifique". Le logiciel à élaborer est défini comme "un programme qui, à la différence des logiciels standard ou progiciels ou des logiciels d'exploitation, est réalisé pour la satisfaction des besoins particuliers et singuliers d'un utilisateur-client" (Lamy, n°298).

En termes contractuels, la distinction s'opère sur le critère de **détermination de l'objet**. Le produit standard est défini indépendamment du contrat, ce dernier ne servant qu'à préciser les modalités de l'échange (prix, délais, nature des droits). Pour le produit spécifique se pose de plus dans le contrat le problème de détermination de l'objet. Cette tâche sera dévolue au cahier des charges, avec les difficultés que l'on connaît.

Pour les progiciels, le contrat prend la forme de la concession d'un droit d'utilisation ; le client n'est pas propriétaire et ne peut pas adapter le produit³.

Dans le quatrième type de contrat -"fourniture clé en main"- les prestations peuvent associer la prise en charge du matériel, la fourniture de progiciels et la **réalisation de logiciels spécifiques**, le point important étant que ces différents éléments forment un tout indissociable et sont évalués globalement : c'est la prestation d'intégration de systèmes qui est au coeur du contrat.

2.3 Les difficultés dans la formation des contrats

Deux difficultés particulières apparaissent dans la formation des contrats, la première se posant de façon plus aiguë pour certaines catégories de logiciels et de prestations, la seconde étant commune à toutes les transactions :

1. La détermination de l'objet

Nous avons vu que ce problème se pose essentiellement en cas de **logiciel spécifique** ; par ailleurs, il concerne surtout les prestations de réalisation de logiciel et de fourniture de système informatique. Dans tous les cas, la charge de la détermination de l'objet est imposée *conjointement* aux deux parties :

"le client doit exprimer ce dont il a besoin ; le fournisseur doit l'assister dans la formulation, si nécessaire l'interroger encore, afin de pouvoir proposer des solutions exhaustives et adéquates, adaptées à ce que souhaitait, explicitement, voire implicitement, l'utilisateur." (Lamy, n°384)

L'imprécision ou l'inadaptation au besoin dans la détermination de l'objet du contrat reste une source majeure de contentieux dès que la prestation n'est pas standard⁴. Il est en revanche plus difficile d'invoquer cette clause dans le cas de progiciel : le client a la

³. Cette incapacité est légale et également technique (le client ne dispose généralement pas des sources des programmes).

⁴. On pourra consulter à ce propos l'ouvrage de Lamberterie (1983) qui présente une analyse de la jurisprudence ainsi que la revue Expertises des systèmes d'information pour un examen de cas plus récents.

possibilité d'observer directement le produit avant la signature du contrat et les tribunaux lui attribuent généralement la charge d'évaluer si celui-ci est conforme à ses besoins⁵.

2. La question du consentement des parties (tous types de contrats)

La difficulté réside ici dans l'*inégalité des situations* entre fournisseur et client, inégalité souvent reconnue par la jurisprudence :

"Il est commun de souligner, dans les relations informatiques, et en raison de la complexité technique de certains de ses aspects, que les partenaires sont bien souvent en situation inégale. Le fournisseur, spécialiste, peut aisément circonvenir l'utilisateur profane ; ils ne parlent pas toujours le même langage." (Lamy, n°333)

C'est cette même inégalité des situations entre un fournisseur, réputé compétent, et un client, réputé profane, qui a généré des obligations originales pour les parties. Ces obligations, attestées par la jurisprudence (Lamberterie, 1983, p.12 et s.), visent principalement la phase de mise en place d'une solution informatique, dans laquelle est définie la prestation d'informatisation.

2.4 Les obligations des parties dans les contrats d'informatique

Nous récapitulons dans le tableau 5.2 les obligations des parties en cherchant à préciser leur fondement. Ces obligations ne concernent pas le contrat d'assistance technique. Elles sont modulables en fonction du type de logiciel (spécifique, standard) et des compétences des parties : les obligations du fournisseur sont ainsi allégées en cas de fourniture d'un progiciel (moins de problèmes dans la détermination de l'objet) ou de compétence informatique établie du client (moins d'inégalité dans la relation).

⁵. Voir par exemple l'affaire Sofresa c. Mac Cormack et Dodge et Fiduciaire de France. Expertises des systèmes d'information. p.31 et s.)

Nature de l'obligation	Fondements
<p>Pour le <u>fournisseur</u> :</p> <ul style="list-style-type: none"> - <i>de renseignement</i> : donner une information objective, neutre, complète - <i>de mise en garde</i> : avertir des risques liés à l'informatisation - <i>de conseil</i> : conseiller l'utilisateur. proposer une solution conforme à ses besoins <p>Pour le <u>client</u> :</p> <ul style="list-style-type: none"> - <i>d'informer</i> : exprimer ses besoins (souvent dans un cahier des charges) - <i>de participer</i> : s'informer correctement sur l'informatique ; s'impliquer dans le projet ; assumer ses choix ; dialoguer avec le fournisseur 	<ul style="list-style-type: none"> - Les décisions de jurisprudence cherchent à compenser l'<u>asymétrie d'information</u> entre un professionnel maîtrisant "une technique obscure et mystérieuse" et son client - Elles cherchent également à instaurer et protéger le rapport de <u>confiance</u> entre les parties - Il est également mis en évidence que la mise en place d'une solution informatique ne peut être que le résultat d'une <u>action conjointe</u> du fournisseur et du client.

Tableau 5.2. Les obligations des parties dans les contrats d'informatique

III. Synthèse des caractéristiques du produit et de la transaction

3.1 Caractéristiques du produit

A partir des différents éléments de l'analyse juridique, nous retenons trois caractéristiques du logiciel qui nous semblent susceptibles de jouer sur les transactions.

1. la difficulté d'évaluation du bien

Cette difficulté résulte à la fois de l'immatérialité de l'objet et de l'immatérialité des résultats produits. Elle se traduit en droit des contrats par la question de la détermination de l'objet. En cas de logiciel spécifique, l'évaluation du bien est forcément repoussée dans le temps ; en cas de progiciel, l'évaluation est possible (mais pas forcément facile) en phase pré contractuelle.

2. l'existence d'un régime de droits de propriété "faible",

Dans de nombreuses situations, les entreprises utilisent des logiciels qu'elles ne possèdent pas : c'est le cas pour les progiciels mais souvent également pour les logiciels spécifiques développés en sous-traitance par une SSII. La possession des logiciels devient donc de plus en plus l'apanage des professionnels du secteur informatique ; protéger ses droits de propriété sur un logiciel est de toute façon difficile, tant légalement (voir les procès qui opposent les géants du logiciel) que pratiquement (il est très difficile d'empêcher toute copie illicite d'un programme par un professionnel informaticien). Le droit d'usage, assorti de garanties, tend donc à remplacer le droit de propriété.

3. le caractère de production jointe

L'adaptation au contexte d'utilisation (machines, organisation, environnement humain) est déterminante du succès d'une transaction portant sur définition du logiciel. Le résultat est donc obtenu par l'action conjointe du fournisseur et du client ; en cas d'échec ou de problème, il est souvent très difficile de déterminer la part de responsabilité attribuable à l'une ou l'autre des parties.

3.2 Les caractéristiques de la transaction d'informatisation

De même, nous pouvons identifier quatre caractéristiques de la transaction d'informatisation.

1. C'est une transaction impliquant souvent une asymétrie d'information

Même si l'image d'un utilisateur qui serait totalement profane peut actuellement prêter à sourire, il reste que le non-professionnel sera souvent démuné en face d'un développement impliquant la maîtrise d'outils technologiques nouveaux et sophistiqués. La gestion de grandes bases de données a présenté ces caractéristiques depuis les dix dernières années ; la maîtrise des réseaux, des Ateliers de Génie Logiciel, de la programmation objet ou des systèmes experts pose également ce type de problèmes.

2. Le facteur temps joue un rôle déterminant

Les contrats de réalisation de logiciel spécifique ou de fourniture de système informatique s'étalent sur des délais souvent assez longs (plusieurs mois, voire plusieurs années). Bien que la tendance soit à chercher à réduire ces délais (en améliorant la productivité ou en "découpant" le projet en phases distinctes faisant chacune l'objet de contrat) la relation contractuelle s'inscrit dès le départ dans une perspective de moyen ou long terme. L'échange "immédiat" ne se produit que dans le cas de logiciels outils (du type des tableurs). Même dans le cas de progiciels, il y aura souvent un temps de mise en place et de formation. Ce temps peut être long dans le cas de progiciels dits très "paramétrables" (plusieurs mois) ; le produit fini est alors la conjonction des fonctionnalités originelles du progiciel et des règles spécifiques à l'entreprise paramétrées dans le produit.

3. Elle requiert une forte coopération des parties prenantes

Cet aspect est souligné à plusieurs reprises et est à rapprocher du rôle du facteur temps précédemment évoqué : c'est une coopération dans la durée, qui va bien au-delà d'un acte ponctuel de vente, qui est nécessaire à la bonne exécution du contrat. Cette caractéristique est renforcée dans le cas de logiciels spécifiques du fait des particularités du mode de développement et de la fréquence des adaptations en cours de projets, au moins dans les premières étapes.

4. Elle peut présenter un niveau de risque élevé

Le risque peut être lié à l'objet du contrat : la jurisprudence a depuis la célèbre affaire Flammarion c. IBM (J.C.P. 74, I, 17603) appliqué l'obligation de mise en garde au domaine informatique. Au-delà du fait savoureux que cette obligation avait été initialement dégagée relativement à la commercialisation des objets dangereux, on trouve ici la reconnaissance du risque associé à ce type de transaction. Ce risque repose sur les "difficultés inhérentes au démarrage des applications informatiques", sur la "fréquente imprécision des délais" (Lamy, n°566).

Le risque peut également être lié aux acteurs : c'est une situation dans laquelle un ensemble de comportements sinon frauduleux, du moins non conformes à l'établissement de relations de confiance ont pu être constatés et réprimés. Ces comportements opportunistes peuvent d'ailleurs être le fait du fournisseur, qui cherche à profiter de son avantage d'expert, mais également celui du client qui cherche à se soustraire à ses obligations et n'apporte pas une réelle volonté de coopération. Les facteurs de coopération obligatoire et de délai dans l'achèvement du contrat fondent objectivement une situation propice à cet opportunisme des acteurs.

Nous avons dressé ici un portrait-type des attributs du logiciel et des transactions. Certains logiciels applicatifs, ou certaines transactions peuvent évidemment faire preuve de caractéristiques différentes : par exemple, si le besoin est très structuré et si le cahier des charges est assorti d'un prototype, la difficulté d'évaluation du bien peut être nettement réduite. Nous avons indiqué dans le tableau 5.3 quelques exemples de facteurs pouvant jouer sur les attributs du produit ou des transactions .

	Valeur de la caractéristique	Facteurs pouvant modifier la valeur de la caractéristique
<u>Attributs du logiciel</u>		
difficulté d'évaluation du bien	forte	Existence d'un prototype ; développement spécifique correspondant à la modification d'une application existante
dépendance au contexte d'utilisation	forte	Peu d'utilisateur impliqués ; procédures d'utilisation relativement standard
caractère de production jointe	très fort	Application des techniques ; la difficulté ne réside pas dans l'analyse des besoins mais dans le choix des technologies appropriées.
<u>Caractéristiques de la transaction</u>		
asymétrie d'information potentielle	forte	bonne connaissance technique de la part du client et bonne maîtrise du processus de développement
délais longs	fort	Identification de sous-projets ; découpage contractuel.
coopération des parties	très forte	Cahier des charges extrêmement précis ; définition complète de l'objet en démarrage de contrat
risque potentiel	fort	Etablissement de procédures de contrôle et de suivi

Tableau 5.3. Facteurs jouant sur les caractéristiques du produit ou de la transaction

Section 2 - L'apport de la théorie de l'agence à notre travail

La théorie de l'agence permet d'éclairer d'un jour particulier la problématique d'externalisation, en posant clairement la question du pilotage et du coût du contrôle de la relation contractuelle. Elle a pour nous l'intérêt de proposer une modélisation très élaborée de la situation contractuelle, et donc de nous aider à comprendre et analyser celle-ci.

La théorie de l'agence s'intéresse potentiellement à toutes les "relations d'agence", ces relations ayant été définies par Jensen et Meckling comme **"un contrat dans lequel une (ou plusieurs) personne(s) [le principal] a recours aux services d'une autre personne [l'agent] pour accomplir en son nom une tâche quelconque, ce qui implique une délégation de nature décisionnelle à l'agent"** (1976, p.308).

Dans la relation contractuelle, le principal est donc à la fois le prescripteur et l'utilisateur de la ressource ou du bien, l'agent est le producteur de la ressource ou du bien.

Cette théorie forme un ensemble complexe, dans lequel on peut distinguer d'une part la branche "normative", développée principalement par des auteurs économistes, très formalisée et mathématisée et dont l'objet est la définition de modèles de partage du risque et d'élaboration de contrats optimaux entre "agent" et "principal", et d'autre part la branche "positive", davantage orientée vers des préoccupations organisationnelles, où l'on cherche à identifier et analyser, à l'aide des concepts d'agence, les situations de conflit entre agent et principal ainsi que les structures de pilotage susceptibles de limiter ou de résoudre ces conflits. Le champ d'application de la théorie est particulièrement vaste, comme en témoigne une gamme de travaux qui concernent aussi bien les contrats d'assurance, les conflits d'intérêt entre managers et actionnaires, le lien entre patient et médecin ou encore, dans les organisations, la relation entre supérieur hiérarchique et subordonné. Pour une revue de la littérature d'agence, nous renvoyons le lecteur aux synthèses de Charreaux (1987), Eisenhardt (1989), Walker (1989) et Baiman (1990). Pour notre part, nous n'en verrons que les aspects directement liés à notre recherche.

I. Les hypothèses et concepts fondamentaux

Les hypothèses et le champ d'application de la théorie de l'agence ont été synthétisés par Eisenhardt (1989) dans le tableau suivant (p.59) :

Idée principale	Les relations principal-agent doivent correspondre à une organisation efficiente des coûts d'obtention de l'information concernant la transaction et de partage du risque
Unité d'analyse	Contrat entre le principal et l'agent
Hypothèses comportementales	Recherche de l'intérêt personnel Rationalité limitée Aversion au risque
Hypothèses organisationnelles	Conflits d'objectifs partiels entre participants au contrat Le critère de résultat est l'efficience Asymétrie de l'information
Hypothèses sur la nature de l'information	L'information est un bien qui peut être échangé
Problèmes contractuels	Problèmes d'agence (<i>moral hazard</i> et sélection averse) Partage du risque
Champ d'application	Relations dans lesquelles le principal et l'agent ont des objectifs partiellement différents et ne sont pas neutres au risque

Tableau 5.4. Les hypothèses et le champ d'application de la théorie de l'agence

Source : Eisenhardt, 1989

La théorie de l'agence comprend un ensemble d'hypothèses, qui sont autant de postulats sur le comportement des acteurs et le fonctionnement organisationnel, et dont beaucoup sont proches de la théorie économique classique. Les acteurs cherchent à optimiser leurs choix en suivant leur propre intérêt. La branche normative de la théorie de l'agence s'appuie sur une

rationalité économique classique en univers probabilisé⁶ : Baiman (1990) définit ainsi les acteurs comme soumis à l'incertitude environnementale mais capables de prendre en compte et d'évaluer les probabilités d'occurrence associées aux différentes évolutions possibles. La branche positive semble, elle, s'appuyer davantage sur une rationalité limitée telle que l'a définie Simon (1958).

Dans les relations d'agence, on est placé en situation à la fois d'asymétrie d'information, car le principal se trouve dans l'incapacité d'observer complètement le comportement de l'agent et également d'incertitude, car le résultat dépend concurremment des objectifs exprimés par le principal, des actions choisies et exécutées par l'agent en fonction de ses intérêts propres et du contexte - encore incertain - dans lequel la relation contractuelle s'insère. Le principal attache donc une valeur à tout système d'information qui permettrait de réduire d'une part le risque comportemental lié à l'agent et d'autre part l'incertitude contextuelle due aux paramètres d'environnement non maîtrisés et/ou incertains. On voit apparaître ici clairement le lien de filiation entre la théorie de l'agence et la théorie de l'information (*Information Economics*), dont l'objectif principal est de déterminer la valeur d'un système d'information⁷. En théorie de l'agence, la réflexion sur le système d'information va permettre de définir différents types de contrats selon l'information disponible : on distinguera ainsi les contrats appuyés sur une observation du comportement de l'agent tout au long de la relation (*behavior-based contracts*) des contrats appuyés sur l'observation *a posteriori* du résultat (*outcome-based contracts*). Un autre aspect majeur de la théorie est d'analyser les risques comportementaux liés à la situation d'agence et à l'asymétrie d'information.

⁶. Pour une définition de l'acteur rationnel voir par exemple Kassouf (1970, p.7, ainsi que les chapitres 3 et 4 du même ouvrage, consacrés à la prise de décision en univers probabilisé).

⁷. Voir par exemple l'excellent ouvrage de Demski (1980) pour une présentation approfondie de la théorie de l'information, Hirshleifer et Riley (1979) pour une synthèse des travaux menés en économie sur le thème de l'incertitude et de l'information et Sundem (1982) pour une analyse des contributions de la théorie de l'information à la recherche en gestion.

Revenons sur les éléments qui forment la relation d'agence. Quatre "conditions nécessaires" doivent être vérifiées pour pouvoir qualifier une relation entre deux acteurs de relation d'agence :

- l'agent a un véritable pouvoir décisionnel sur l'action qu'il engage. Il ne reproduit en aucun cas mécaniquement un comportement dicté par le principal.

- il y a divergence potentielle d'intérêt entre le principal et l'agent. Par exemple, un présupposé classique de la théorie de l'agence est l'aversion à l'effort de la part de l'agent qui va donc chercher -s'il en a l'opportunité - à limiter sa contribution.

- les actions de l'agent jouent sur le résultat produit, mais de façon non mécanique car ces actions s'insèrent dans un contexte d'incertitude sur l'environnement.

- le principal, comme nous l'avons déjà signalé, n'est pas toujours à même d'observer parfaitement le comportement de l'agent.

Dans cette situation, le problème pour le principal est d'élaborer par contrat un système d'incitations et/ou de contrôle qui pousse l'agent à agir dans le sens de l'intérêt du principal et qui établisse un partage du risque acceptable par les deux parties. Eisenhardt indique ainsi que **"le coeur de la théorie principal-agent est l'arbitrage entre (a) le coût d'évaluation du comportement de l'agent et (b) le coût d'évaluation du résultat et de report du risque sur l'agent"** (1989, p.61).

Les apports de la théorie de l'agence à notre problématique d'externalisation nous semblent pouvoir être classés en trois groupes que nous détaillons ci-après :

1. la théorie de l'agence permet de mettre en évidence et de formaliser certains risques comportementaux pré- et post-contractuels
2. les auteurs ont identifiés un ensemble de coûts supplémentaires, dits "coûts d'agence", qui naissent de la situation de délégation décisionnelle
3. la théorie donne des éclairages sur la nature des contrats, les possibilités de pilotage et les facteurs de motivation adaptés à différentes situations.

Précisons avant d'aborder ces différents points que si un ensemble de travaux en théorie de l'agence ont pris pour cadre une organisation (ses actionnaires, ses managers, ses employés), l'application de la théorie ne se limite pas à ce seul cadre. En particulier, la situation d'agence peut prendre effet dans une relation liant des membres d'organisations différentes. Ce n'est pas tant le lieu de la relation (interne ou externe) que les modalités de décision et de contrôle de l'exécution de cette relation qui importent.

II. Les risques comportementaux pré- et post-contractuels

La théorie de l'agence identifie deux risques comportementaux, l'un pré-contractuel -la sélection adverse (*adverse selection*) - et l'autre post-contractuel, connu sous le nom de *moral hazard* (risque moral).

2.1 La sélection adverse

La sélection adverse intervient quand l'agent donne une représentation inexacte de ses aptitudes en phase pré contractuelle. Le principal se trouve dans l'incapacité de vérifier ces compétences, et cette incapacité peut subsister pendant l'exécution du contrat (Eisenhardt, 1989). Cette situation peut conduire à la sélection d'un agent non efficient qui met en danger l'obtention du résultat. Le risque de sélection adverse se pose dans tous les marchés où la qualité du produit -ou celle du fournisseur - ne peut pas être observée facilement (Hirshleifer et Riley, 1979). Dans de tels contextes, le fournisseur offrant (à un coût non négligeable pour lui-même) un produit de qualité supérieure au marché se verra non récompensé et donc disparaîtra à terme du marché.

Ce problème a été traité par Akerlof (1970) dans un article célèbre sur la présence des *lemons* (voitures de mauvaise qualité) dans le marché de l'occasion. **La sélection adverse naît de l'incertitude sur la qualité du bien** : à partir du moment où l'acheteur est incapable de faire la différence *a priori* entre une bonne et une mauvaise occasion, l'une et l'autre seront vendues au même prix. La malhonnêteté de certains vendeurs lèse leurs cocontractants ; elle a

également un impact défavorable sur le marché en défavorisant les vendeurs honnêtes et en limitant le volume global des échanges. Akerlof indique quelques palliatifs au problème d'incertitude sur la qualité :

- la mise en place par le vendeur de garanties contractuelles de qualité,
- l'existence de marques qui fonctionnent comme un signal de qualité,
- l'existence de chaînes (par exemple les chaînes de restauration) qui assurent la même fonction qu'une marque,
- la délivrance de certificats.

L'objectif de ces dispositifs est d'émettre des signaux de qualité qui visent à rétablir la confiance et, pour l'agent "de bonne qualité", à indiquer au principal sa compétence. Bien évidemment, l'émission de ces signaux correspond à un coût supplémentaire encouru par les institutions qui régulent le marché (coût d'établissement des labels) et par les agents (coût d'obtention du label).

Cette problématique a été appliquée au logiciel par cinq chercheurs anglais dans une étude très intéressante sur la valeur perçue d'une assurance qualité (Jobber et al., 1989). Les objectifs de l'étude étaient d'identifier les problèmes auxquels étaient confrontés les acheteurs de logiciels applicatifs de gestion et de déterminer si l'existence d'un certificat de qualité pourrait résoudre, au moins partiellement, ces problèmes.

Parmi les problèmes identifiés par les auteurs est apparu celui de la faible qualité de certains logiciels ou d'éléments associés (documentation), spécialement dans le cas de logiciels spécifiques et/ou de logiciels produits par des SSII de petite taille. Les résultats ont montré un intérêt réel pour un certificat de qualité (75% des répondants choisiraient de préférence un produit bénéficiant du certificat, parmi lesquels 79% sont prêts à payer davantage pour obtenir ce service). Bien que cette étude n'ait porté que sur un nombre très faible de cas, elle est cohérente avec notre analyse du logiciel applicatif de gestion comme un bien dont la qualité est difficile à attester *a priori*, condition qui forme des marchés propices au phénomène de sélection adverse.

Dans le domaine des progiciels, on retrouve les mêmes difficultés. C'est pourquoi certaines sociétés de services ont cherché à faire certifier leurs progiciels comptables par des cabinets d'expertise comptables reconnus⁸.

2.2 Le *Moral Hazard*

Le problème du *moral hazard* est de nature post-contractuelle. Il fait référence au fait que l'agent peut choisir délibérément de ne pas exécuter le contrat comme prévu mais au contraire modifier son comportement, dans la recherche de son propre intérêt.

Ce comportement de l'agent est rendu possible par deux éléments :

- l'impossibilité d'établir des contrats complets, compte tenu de l'incertitude qui pèse sur les états futurs du monde ainsi que sur les règles de causalité entre les actions de l'agent et les résultats obtenus,
- l'incapacité pour le principal d'observer complètement le comportement de l'agent.

Dans ce cas, même si l'agent a été bien sélectionné *a priori*, la recherche normale de son intérêt peut le conduire à jouer de la situation à son profit en cours d'exécution du contrat.

Un cas particulier de la relation d'agence relevé par Jensen et Meckling (1976), se référant aux travaux de Alchian et Demsetz (1972), est la situation dans laquelle la production du résultat implique un effort fourni conjointement par plusieurs personnes. Dans de telles situations, le risque de *moral hazard* est élevé si l'on ne peut pas mesurer l'impact de l'effort individuel sur le résultat commun, chaque agent ayant l'opportunité de "tricher" en limitant son effort. Ce risque ne pourra être limité que si l'on met en place des mécanismes d'observations du comportement de l'agent, qui engendrent des coûts supplémentaires.

⁸. Nous avons mené en 1987 une telle certification sur un progiciel Nixdorf dans le cadre d'activités salariées auprès d'un grand cabinet de conseil et d'expertise français.

Deux caractéristiques des applications forment une situation propice au risque de *moral hazard* :

- l'existence d'adaptations de la conception en cours de projet ; il est souvent impossible de déterminer au départ une fois pour toutes les fonctionnalités. Celles-ci peuvent de plus évoluer en réaction aux modifications de l'environnement,
- le caractère de production jointe entre utilisateurs et informaticiens. En cas d'échec, il est difficile de faire le partage des responsabilités.

III. Les coûts d'agence

Des coûts spécifiques d'agence ont été mis en évidence par Jensen et Meckling (1976), qui les classent en trois groupes :

1. la perte résiduelle (*residual loss*), que Charreaux (1987) analyse comme un coût d'opportunité et qui correspond à une "sous-optimisation" du marché due aux divergences d'intérêt entre principal et agent.

2. les coûts de pilotage de la relation (*monitoring*), encourus par le principal, et parmi lesquels on peut distinguer :

- les coûts d'établissement d'un système d'incitations adapté qui favorise un comportement de l'agent conforme aux intérêts du principal,

- les coûts de surveillance de la relation contractuelle, de manière à pouvoir réprimer les comportements non conformes au contrat. Jensen et Meckling citent les coûts associés aux méthodes d'audit, à la mise en place de systèmes formels de contrôle et aux procédures budgétaires. Ce sont des coûts liés à la mise en oeuvre d'un système d'information de contrôle.

3. les coûts d'obligation (*bonding costs*), encourus par l'agent. Ces coûts incluent des garanties contractuelles, l'établissement de pénalités et des limitations au pouvoir de décision de l'agent.

Nous classons également parmi les coûts d'obligation les coûts d'émission des signaux prouvant au principal la bonne qualité de l'agent, auxquels nous avons déjà fait référence. Certains d'entre eux sont encourus en phase pré contractuelle (obtention d'un diplôme,

publicité pour affirmer une marque), mais également lors de l'exécution du contrat (respect de procédures pour attester du sérieux du travail réalisé). Ce sont en fait tous les coûts liés à l'existence et au maintien de la réputation de l'agent⁹.

IV. Le pilotage de la relation d'agence

Le problème du pilotage de la relation d'agence pose deux questions :

1. comment établir les facteurs de motivation (*incentives*) pour l'agent de manière à faire coïncider le plus possible son intérêt avec celui du principal ?
2. quel système de contrôle mettre en place pour identifier et réprimer les comportements préjudiciables de la part de l'agent ?

A partir de ce problème de pilotage, Eisenhardt (1989) a développé un ensemble d'hypothèses parmi lesquelles :

Hypothèse 1 : L'agent est davantage susceptible d'agir dans le sens des intérêts du principal quand le *contrat prévoit des récompenses liées au résultat produit*.

Hypothèse 2 : L'agent est davantage susceptible d'agir dans le sens des intérêts du principal quand celui-ci a la *possibilité d'obtenir des informations sur le comportement de l'agent*.

La première hypothèse met l'accent sur les facteurs de motivation, la seconde sur le système de contrôle.

Eisenhardt part du principe qu'il y a deux manières de rémunérer les agents - sur leur comportement ou sur leur résultat - et distingue ainsi les contrats dans lesquels la récompense de l'agent est liée au résultat obtenu (*outcome-based contracts*) de ceux où elle est liée au comportement de l'agent (*behavior-based contracts*). Dans la première catégorie de contrats, la rémunération de l'agent est calculée une fois le résultat obtenu, à partir de règles déterminées à l'élaboration du contrat. Savoir comment l'agent est parvenu au résultat, quelle est la somme

9. Pour une présentation de la "théorie de la réputation" voir Holmstrom et Tirole 1989, p76 et s.)

d'efforts qu'il a engagés est inutile pour déterminer la rémunération. Un contrat basé sur le comportement comprend en revanche un ensemble de règles et procédures auxquelles l'agent devra se conformer. Il est à noter que l'on peut observer ces deux types de contrats pour des transactions portant sur les logiciels ; ce sont respectivement le contrat au **forfait** (qui correspond à un résultat donné) et le contrat en **régie** (où le sous-traitant travaille sur place, sous la direction de l'entreprise qui peut observer son comportement).

Eisenhardt identifie ensuite un ensemble de critères qui conduiront à retenir l'une ou l'autre forme contractuelle, parce que plus efficiente dans une situation donnée. Ces critères sont récapitulés dans le tableau suivant :

Critères de contingence liés au choix de la forme contractuelle	Situation de meilleure efficacité d'un contrat :	
	<i>behavior-based</i>	<i>outcome-based</i>
Incertitude sur le résultat	forte	faible
Aversion au risque (de la part de l'agent)	forte	faible
Aversion au risque (de la part du principal)	faible	forte
Conflits d'objectifs entre agent et principal	faibles	forts
Caractère programmable de la tâche	fort	faible
Possibilité de mesurer le résultat	faible	forte
Durée de la relation	longue	courte
Systèmes d'information de contrôle	développés	limités

Tableau 5.5. Le choix entre une rémunération contractuelle sur la base du comportement ou en fonction du résultat obtenu

D'après Eisenhardt (1989)

Le facteur d'allocation du risque est central dans la théorie de l'agence et correspond aux trois premiers critères évoqués. Les autres critères posent le problème d'**observation** et de **mesure** de la qualité du comportement et/ou du résultat.

Si les conflits d'objectifs sont faibles, le principal pourra compter *a priori* sur un "bon" comportement de l'agent, et l'évaluation de ce comportement sera moins nécessaire. Le caractère programmable d'une tâche (référence au fait que le processus est connu et peut être défini à l'avance par des règles) rend l'observation du comportement plus aisée. De même, dans une relation d'agence de longue durée, le principal finit par "connaître" l'agent et appréhende mieux son comportement. Toutes ces situations sont favorables à la mise en oeuvre de contrats récompensant le comportement, de même est favorable à ces contrats la possibilité de mettre en oeuvre à coût supportable un système d'information de contrôle pendant l'exécution de la tâche (budgets, reporting).

A l'inverse, si le résultat peut être facilement évalué (et en cas de production jointe, si l'apport marginal de chacun au résultat peut être mesuré), les contrats liant la rémunération au niveau de résultat atteint sont plus efficaces. Ils le sont également en cas de conflits potentiels élevés (divergence des buts) et de processus difficilement auditable (tâche non programmable).

Conclusion : apports et limites de la théorie de l'agence pour notre recherche

La théorie de l'agence nous a permis :

- d'explicitier certains risques comportementaux liés aux contrats (*adverse selection* et *moral hazard*) ; nous avons pu établir que ces risques sont présents dans la relation contractuelle qui nous intéresse,

- d'identifier des "coûts d'agence" (coûts d'opportunité, coûts de pilotage, coûts d'obligation) ; les coûts de pilotage des projets informatiques sont très élevés et nous avons vu que les fournisseurs de logiciel encourent des coûts pour maintenir leur réputation et celle de leurs produits,

- de lier selon une approche contingente le mode de pilotage choisi (*outcome-based* ou *behavior-based*) à un ensemble de facteurs d'allocation du risque et de facilité d'observation et de mesure du comportement et du résultat. Ces deux modes de pilotage sont couramment utilisés pour le développement d'applications (évaluation du comportement en cas de réalisation interne et de régie ; évaluation du résultat en cas de forfait) .

Appliquée à notre domaine, la théorie de l'agence nous suggère les hypothèses suivantes :

1. le choix entre les deux modes de pilotage dépend principalement de la possibilité de mettre en oeuvre au moindre coût l'un ou l'autre des deux modes de pilotage (possibilité de contrôle du résultat et possibilité de contrôle du comportement)

2. la possibilité de contrôle du résultat est meilleure si le projet est bien défini *a priori* (cahier des charges complet, prototypage),

3. la possibilité de contrôle du comportement est meilleure si la relation entre les deux parties est établie sur une durée longue ; elle nécessite la mise en oeuvre de systèmes de contrôle coûteux.

Le tableau 5.6. présente la synthèse de cette approche.

En revanche, et c'est là la principale difficulté pour nous, la théorie de l'agence ne permet pas d'évaluer directement les mérites comparés de l'organisation, du marché ou encore d'une relation personnalisée avec un sous-traitant. En effet, l'une des hypothèses principales de la théorie de l'agence est de considérer qu'il n'y a pas de différence profonde de nature entre une relation médiatisée par le marché (le contrat) et une relation médiatisée par l'organisation (la relation de l'employé à son supérieur hiérarchique ou aux autres employés). C'est d'ailleurs l'un des apports principaux de cette théorie d'introduire la vision de la firme comme un "noeud de contrats" (Jensen et Meckling, 1976, p.310) et d'appliquer une approche homogène à des situations organisationnelles et aux relations de marché.

Il reste que l'on peut s'interroger, dans le cadre d'une relation de type client-fournisseur, sur les capacités relatives de la firme, du marché ou d'autres modes contractuels à limiter les coûts d'agence. Il s'agit alors d'évaluer les aptitudes respectives de ces différents modes pour :

- définir des facteurs d'incitations (*incentives*) efficaces,
- mettre en oeuvre un système d'information permettant d'observer et d'évaluer le résultat d'une relation contractuelle,
- mettre en oeuvre un système d'information permettant d'observer et d'évaluer le comportement de l'agent dans cette relation,
- limiter le besoin de surveillance, soit par l'action des facteurs d'incitations, soit en assurant une convergence *a priori* et/ou une relation de confiance entre les parties.

Pour chercher des réponses à ces questions, nous nous tournons dans le chapitre suivant vers une autre théorie contractuelle, partageant de nombreux points communs avec la théorie de l'agence, mais dont l'objectif est de déterminer les vertus comparées de la firme et du marché pour gérer les transactions : la théorie des coûts de transaction de Williamson.

Principaux éléments liés à la théorie de l'agence	Application à l'activité de développement d'applications	Difficultés et limites
<p>L'externalisation est étudiée sous l'angle de la relation contractuelle.</p> <p>La théorie de l'agence permet :</p> <ul style="list-style-type: none"> - de formaliser les risques comportementaux pré et post-contractuels (sélection adverse et moral hazard), - d'identifier des coûts d'agence, principalement coûts de pilotage et coûts d'obligation, - de distinguer deux modes principaux de pilotage de la relation <ul style="list-style-type: none"> . par l'examen du résultat (<i>result-based contracts</i>) . par l'examen du comportement (<i>behavior-based contracts</i>). 	<p>Les caractéristiques juridiques du logiciel et de la transaction de développement spécifique forment un contexte propice à la formation des coûts d'agence :</p> <ul style="list-style-type: none"> - asymétrie d'information, - difficulté d'évaluation du résultat, - difficulté d'établissement de contrats complets (modifications en cours de projet), - caractère de production jointe (utilisateurs-informaticiens). <p>Les deux modes de pilotage identifiés par la théorie de l'agence correspondent aux contrats utilisés dans le développement d'applications</p> <ul style="list-style-type: none"> - observation du comportement (contrats de régie ou réalisation interne), - observation du résultat (contrat au forfait). 	<p>L'évaluation du résultat en fin de projet présente des risques forts pour l'organisation.</p>

Conditions de mise en oeuvre de l'externalisation	Variables à prendre en compte dans la décision d'externalisation
<p>Possibilité d'observer le comportement (connaissance du processus, possibilité d'observer et de mesurer le travail fourni).</p> <p>Possibilité d'observer le résultat (critères d'évaluation du logiciel spécifique, évaluation <i>a priori</i> ou en cours de projet).</p>	<p>Les variables vont permettre de choisir le mode de pilotage (observation du résultat et observation du comportement) :</p> <ul style="list-style-type: none"> • Complétude de la définition du résultat (qualité du cahier des charges, prototypage) • incertitude sur la qualité • durée de la relation.

Tableau 5.6 Synthèse de l'approche de la théorie de l'agence

Chapitre 6 :
L'étude de la relation contractuelle -
les apports de la théorie des coûts de transaction

Introduction

La "théorie des coûts de transaction" est issue des travaux de Williamson (principalement 1975 et 1985) ; c'est une théorie jeune et en évolution mais qui a déjà suscité un volume très important de travaux, dont beaucoup se situent en dehors du champ des systèmes d'information mais ont un rapport étroit avec notre objet de recherche. Elle a été appliquée avec profit à des problématiques de choix de ressources dans différents domaines, comme les conditions d'externalisation d'une force de vente (Anderson et Weitz, 1986), l'efficacité comparée de la gestion interne et de l'externalisation d'activités comme la recherche et développement, la vente et les prestations de services dans le secteur informatique (Mosakowski, 1991), le choix des composants à réaliser en interne, à acheter ou à sous-traiter dans une procédure de production (Masten et Meehan, 1991), ou encore l'intégration verticale dans le secteur chimique (Lieberman, 1991).

La problématique centrale des travaux de Williamson est l'arbitrage entre ressource interne et externe. Pour un échange donné ("transaction"), l'auteur cherche à analyser les mérites respectifs des différentes ressources ("structures de gouvernance") envisageables. Williamson distingue trois structures de gouvernance : le marché (concurrence parfaite entre des fournisseurs extérieurs), la hiérarchie (gestion interne) et les formes contractuelles hybrides (contrats relationnels alliant des mécanismes de concurrence à une relation personnalisée). Selon Williamson, pour une *même* transaction, les différentes structures de gouvernance encourront des coûts de transaction - c'est-à-dire des coûts *de planification, d'adaptation et de pilotage* des tâches - différents. Ces coûts dépendent à la fois de la nature de la transaction et des capacités de la structure de gouvernance retenue.

Envisager l'externalisation du développement des applications dans le cadre de la théorie des coûts de transaction revient à se poser les questions suivantes : les entreprises sont en train d'évoluer pour la gestion de leurs ressources en logiciel applicatif d'un mode d'organisation traditionnel interne (hiérarchique) vers une coordination par le marché ? Si cela est le cas, peut-

on identifier des facteurs objectifs, que ce soit dans la nature des transactions ou dans les capacités respectives des différents modes de gouvernance, qui expliquent cette évolution ?

La première section de ce chapitre est consacrée à la présentation de la théorie. Nous insisterons en particulier sur les travaux récents de Williamson portant sur les capacités respectives des structures de gouvernance.

La seconde section analyse la pertinence de ce cadre théorique pour l'analyse du problème d'externalisation du développement d'application.

Section 1 - La théorie des coûts de transaction

Depuis une vingtaine d'années, et sous l'impulsion continue de O. Williamson, s'est développé le courant de recherche connu sous le nom de "théorie des coûts de transaction" (que nous abrègerons parfois, dans un souci de rapidité de lecture, en TCE - *Transaction Cost Economics*). Se référant aux travaux des années 1930 en économie (Coase, 1937, Commons, 1934), en droit (Llewellyn, 1931) et en théorie des organisations (Barnard, 1938), Williamson cherche à évaluer les mérites comparés du marché, de la firme et de formes contractuelles intermédiaires pour minimiser le coût de gestion des échanges. Le champ des applications de la théorie des coûts de transaction dépasse largement le cadre du problème qui nous intéresse. On se limitera ici aux principaux éléments de ce modèle, en mettant l'accent sur ceux qui nous semblent particulièrement éclairer notre recherche et en renvoyant le lecteur pour un éclairage plus complet aux excellentes synthèses déjà disponibles¹.

Le modèle de Williamson s'oppose à une vision de la firme restreinte à la seule fonction de production et permet d'étudier d'autres facteurs explicatifs des choix d'externalisation que l'analyse technologique et les coûts de production. L'entreprise peut chercher à minimiser, en optimisant son choix d'approvisionnement, ses "coûts de transaction", c'est-à-dire les coûts *de planification, d'adaptation et de pilotage* des tâches (Williamson, 1985, p.2). Cette étude est comparative : il s'agit d'identifier les mérites respectifs de structures alternatives d'organisation des transactions.

Nous rappellerons tout d'abord les principales définitions et hypothèses de la TCE ; nous nous intéresserons ensuite aux attributs des transactions ; nous terminerons cette présentation de la théorie par l'analyse des caractéristiques des structures de gouvernance.

¹. La littérature produite en théorie des coûts de transaction atteint, malgré sa relative jeunesse, un volume fort impressionnant. Parmi les nombreuses publications de O. Williamson, citons deux ouvrages de référence (1975, 1985). Nous renvoyons également aux synthèses réalisées par différents auteurs : Reix (1979), Joffre (1985), Joffre et Koenig (1985), Brousseau (1989), Garett (1991) et aux très nombreux articles parus notamment dans le "Journal of Law and Economics" depuis une vingtaine d'années.

I. Définitions et rappel des hypothèses

Le mérite reconnu de la théorie des coûts de transaction est de proposer un modèle d'analyse cohérent, différent du modèle technologique, et dont les hypothèses et les paramètres ont été clairement définis par O. Williamson.

1.1 Transactions, structures de gouvernance et coûts de transaction

Au centre de la théorie se trouvent les transactions et les structures de gouvernance de ces transactions (*governance structure*) :

"Une **transaction** intervient quand un bien ou un service traverse une interface technologiquement séparable². Une étape de l'activité se termine et une autre commence" (1981, p.1544). Plus largement, par transaction Williamson entend un échange de bien ou service entre différents acteurs.

La "**structure de gouvernance**" est définie comme : "le cadre contractuel explicite ou implicite dans lequel s'inscrit la transaction" (1981, p.1544). Dans les premières étapes de sa réflexion, Williamson oppose deux modes alternatifs extrêmes de pilotage des transactions, la hiérarchie³ et le marché (1975), avant d'élargir son analyse aux modes intermédiaires qu'il qualifie de formes hybrides (1985). C'est cette approche élargie que nous retiendrons ici.

La transaction peut être opérée harmonieusement, mais des malentendus ou des conflits peuvent également survenir entre les partie-prenantes à l'échange et générer retards, échecs et dysfonctionnements. Ce sont ces imperfections des échanges et les efforts menés pour les éviter qui génèrent les **coûts de transaction**. Ces coûts peuvent intervenir à différents moments de la transaction, *ex ante* ou *ex post* (1985, p.20-22) :

². "when a good or service is transferred across a technologically separable interface".

³. La hiérarchie correspond à l'organisation interne. Nous employons ces deux termes de façon équivalente dans toute la partie consacrée à la théorie des coûts de transaction.

1. Les coûts *ex ante* incluent la rédaction, la négociation et la mise en oeuvre de système de protection des contrats. Les coûts d'élaboration et de négociation peuvent être très élevés si l'on cherche à garantir l'échange *a priori* en élaborant des contrats complets, c'est-à-dire en prévoyant toutes les contingences possibles et les obligations des parties en réponse à chaque contexte particulier (l'alternative à ces contrats complets est la définition de contrats volontairement généraux et incomplets, redéfinis au fur et à mesure des évolutions de la situation, ce qui générera des coûts *ex post* de renégociation).

2. Parmi les coûts *ex post* on trouve les coûts résultant d'une adaptation inefficace aux variations de l'environnement, les coûts de renégociation évoqués précédemment, les coûts d'entretien des structures de suivi du contrat et de résolution des litiges et les coûts d'obligation encourus par les parties.

En pratique, l'éventail des coûts de transaction peut s'avérer très large et ceux-ci sont souvent définis *a contrario*, par ce qu'ils ne sont pas : ce ne sont ni des coûts de production ni des coûts de livraison. En revanche, les honoraires de conseils juridiques, le travail nécessaire à la recherche d'un compromis, les efforts de standardisation et de certification de la qualité des biens échangés, les impôts attachés à certains types de transaction font partie des coûts de transaction (Joffre, 1985). En font également partie les coûts d'élaboration et de maintien du système de pilotage et de contrôle (donc dans le cas d'une transaction internalisée, le coût du contrôle de gestion de l'activité - par exemple sous la forme de suivi de projet - et dans le cas de recours au marché, le coût de pilotage de la sous-traitance - sélection du sous-traitant, réunions de point, renégociations, amendements au contrat).

Le coeur de la théorie des coûts de transaction est l'affirmation selon laquelle **des structures de gouvernance différentes encourrent pour une même transaction des coûts de transaction de niveau variable**. Concernant les structures de gouvernance, Williamson s'oppose ainsi radicalement à la théorie de l'agence en réfutant l'argument selon lequel la firme et le marché ne se distinguent pas sur le plan contractuel (1991a, p.274 et 276). Son objectif est au contraire de définir et de fonder un modèle de correspondance entre attributs des

transactions et structures de gouvernance, c'est-à-dire : "d'associer sélectivement des transactions, qui diffèrent dans leurs attributs, et des structures de gouvernance, qui diffèrent dans leurs coûts et leurs capacités, sur le critère principal d'économie des coûts de transaction" (1991b, p.79).

Appliquée au problème du choix entre ressource interne ou externe, la théorie des coûts de transaction cherche donc à déterminer, en fonction des attributs de la tâche, du produit ou du service pour lequel se pose le choix d'externalisation, quelle est la meilleure solution d'organisation (interne, marché ou formes contractuelles intermédiaires), c'est-à-dire quel est le mode d'organisation qui minimisera la somme des coûts de production et de transaction.

1.2 Les hypothèses comportementales et l'influence de l'environnement

Le schéma des imperfections organisationnelles élaboré par Williamson dès 1975 met en évidence les hypothèses comportementales retenues par l'auteur et l'influence de l'environnement sur les transactions (figure 6.1).

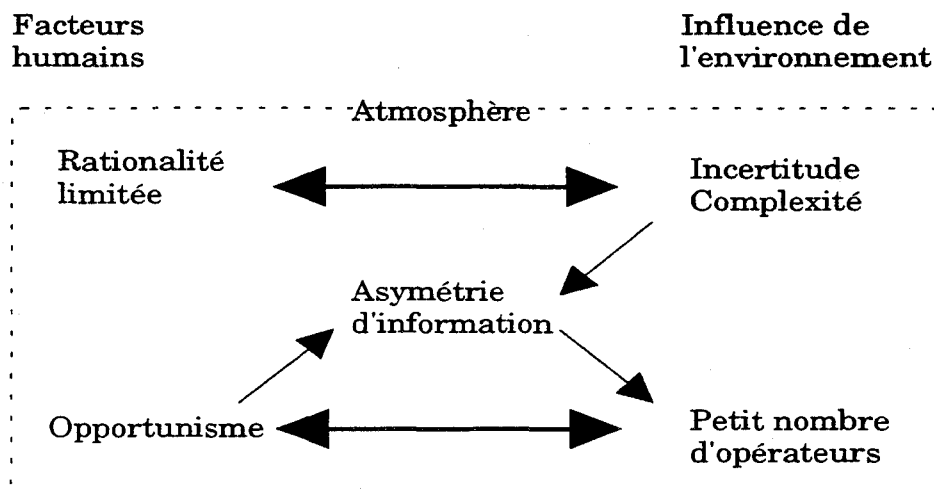


Figure 6.1. Schéma des imperfections organisationnelles

(source : Williamson, 1975)

Suivant Knight dans la nécessité de prendre en compte pour l'étude des organisations "la nature humaine telle que nous la connaissons", Williamson (1989, p.138) s'écarte de l'homo oeconomicus pour brosser le tableau d'un "homme contractuel" opportuniste et dont la rationalité est limitée.

En adoptant le principe de **rationalité limitée**, Williamson se réfère à H.A.Simon (1947) pour lequel les acteurs économiques sont "intentionnellement rationnels", mais sont limités dans l'exercice de la rationalité par leurs capacités cognitives. Il s'éloigne ainsi de la branche normative de la théorie de l'agence et des modèles mathématiques visant à définir des contrats optimaux. De la rationalité limitée naît en effet l'impossibilité de former des contrats complets, c'est-à-dire prévoyant *ex ante* l'ensemble des évolutions possibles et leur traitement. Cette impossibilité est renforcée dans un contexte d'**incertitude** et de **complexité** environnementale. C'est pourquoi, dans l'étude des coûts de transaction, Williamson met particulièrement l'accent sur le traitement *ex post* des contrats et sur la mise en oeuvre des capacités d'adaptation en cours de réalisation des transactions.

Le concept d'**opportunisme** correspond à la forme forte de la recherche de l'intérêt personnel, ce qui peut inclure la pratique du mensonge ou de la fraude (Williamson, 1985, p.47). Ce concept est proche du *moral hazard* de la théorie de l'agence, dont il constitue une généralisation pour inclure l'ensemble des risques contractuels liés à cette forme poussée de la recherche de l'intérêt personnel.

Pour que l'opportunisme puisse s'exprimer, il est néanmoins nécessaire qu'il trouve un contexte favorable, formé par la présence d'un nombre restreint d'opérateurs sur le marché, qui rende difficile la substitution d'un co-contractant, dont l'opportuniste a été décelé, par un autre. Williamson développe cet argument en notant que cette situation de "**petit nombre**" peut se développer au cours d'une transaction, même si au départ la concurrence s'exerce entre de nombreux fournisseurs. C'est la "transformation fondamentale" (1985, p.61) qui intervient quand les co-contractants investissent de manière spécifique dans la transaction et se trouvent ainsi bloqués dans une relation bilatérale. En effet, à moins de pouvoir transférer à faible prix

les investissements réalisés vers d'autres fournisseurs, interrompre la transaction et changer d'opérateur implique de subir une perte (*sunk costs*). Il y a donc des freins très importants à la substitution dès que la transaction implique la mise en oeuvre d'investissements spécifiques.

L'opportunisme peut, de même qu'un environnement incertain et complexe, générer ou renforcer une **asymétrie d'information**. Selon l'hypothèse d'opportunisme, il faut s'attendre en effet à ce que les acteurs des contrats, cherchant à satisfaire leur intérêt propre, "soient malhonnêtes, délivrent une information fautive sur leurs capacités ou leurs préférences, manipulent les données, cachent les problèmes et rendent les transactions confuses" (1981, p.1545). Cette asymétrie conduit par ailleurs à renforcer la situation de petit nombre déjà identifiée. Il faut noter ici que Williamson, contrairement à la théorie de l'agence, ne fait pas de l'asymétrie d'information une hypothèse centrale de son examen des situations contractuelles. Sans l'action conjointe de l'opportunisme, de la rationalité limitée et de la situation de petit nombre, l'asymétrie d'information ne pénaliserait pas les échanges sur le marché (1975, p.33).

A ces différents éléments, Williamson ajoute la prise en considération d'aspects qualitatifs traduisant l'attitude des agents, "l'**atmosphère**". Ce terme fait référence aux influences du contexte économique ou de facteurs sociologiques sur les valeurs et les comportements des agents. Le cadre dans lequel s'inscrit la transaction va déterminer au moins en partie l'atmosphère : ainsi Williamson indique-t-il que le marché encourage principalement les attitudes calculatrices en limitant les engagements aux éléments mesurables formellement inscrits dans le contrat. L'organisation interne, par contraste, forme un meilleur contexte pour des engagements quasi-moraux entre les parties (comme en témoignent par exemple les concepts de réciprocité ou d'équité). Ce thème de l'atmosphère, présent dans les écrits de Williamson en 1975, va connaître une éclipse dans les écrits suivants consacrés principalement à l'analyse des attributs des transactions (*The Economic institutions of capitalism*, 1985). Il réapparaît avec l'analyse récente des caractéristiques des structures de gouvernance (1991a). Il est également très présent dans les problématiques de contrôle qui s'appuient sur les coûts de transaction (Ouchi, 1980).

Les hypothèses comportementales et environnementales faites par Williamson composent le contexte propice au développement de coûts de transaction. Au fur et à mesure que progresse la théorie des coûts de transactions, Williamson s'est attaché à définir les facteurs jouant directement sur les coûts de transaction, en focalisant tout d'abord son analyse sur les attributs qui différencient les transactions, puis en revenant pour les préciser sur les caractéristiques des structures de gouvernance.

II. Les attributs des transactions

Trois attributs distinctifs des transactions vont permettre de fonder un choix économique rationnel du mode d'organisation : la **spécificité des actifs**, la **fréquence** et l'**incertitude** (1985, p.52), auxquels Williamson envisage plus tard d'ajouter un quatrième élément, la **facilité d'évaluation du bien ou du service** (1991b, p.79).

La **fréquence** des transactions pour l'acheteur justifie la mise en place d'une structure spécialisée. Williamson fait ainsi l'hypothèse que seront plus facilement internalisées les transactions répétitives.

L'**incertitude** peut provenir de deux origines. La première source d'incertitude, classique en économie, est celle liée à la non connaissance des états futurs du monde ; mais l'auteur met davantage l'accent sur l'incertitude liée aux comportements⁴ (*behavioral uncertainty*) qui interviendront en réponse à ces évolutions inconnues de la situation. Si une transaction est soumise à une incertitude forte, les limites humaines interdisent la formation de contrats complets (prévoyant tous les états possibles et leur traitement). Des adaptations non prévues interviennent ; l'hypothèse est que les coûts de gestion de ces adaptations sont moins élevés dans la hiérarchie que sur le marché. En particulier, l'opportunisme des acteurs intervenant sur le marché rend toute adaptation très coûteuse.

⁴. On retrouve ici une préoccupation très proche de la théorie de l'agence.

Mais c'est surtout à la spécificité des actifs que Williamson attribue le caractère déterminant pour choisir le mode d'organisation approprié (1991b, p.80). Nous allons donc nous attarder quelques instants sur ce critère.

2.1 La spécificité des actifs

Le critère de spécificité permet d'évaluer : "dans quelle mesure un actif peut être redéployé pour d'autres usages et pour d'autres utilisateurs sans sacrifier de sa valeur productive" (1991a, p.281). **L'hypothèse est que plus une transaction implique de mettre en oeuvre des actifs spécifiques, plus un mode d'organisation interne est approprié (la firme devient plus efficace que le marché).**

Williamson distingue six différents types de spécificité des actifs (1991a) :

- la spécificité géographique (relation où la proximité du fournisseur est déterminante),
- la spécificité physique (machines, composants),
- la spécificité des actifs humains (développement d'expertise ou de savoir-faire),
- les actifs dédiés,
- l'existence d'une marque,
- la spécificité temporelle⁵.

Il est important de noter que la spécificité des actifs se mesure pour un fournisseur *lambda* du marché et correspond pour lui à la question : "Dois-je faire des investissements spéciaux, développer un savoir-faire distinct, élaborer une image de marque, changer significativement mon organisation afin de respecter des délais particulièrement serrés, ... pour répondre à l'offre d'un client particulier ?"

⁵. Ce dernier type de spécificité fait référence aux situations où le délai (de livraison) a un caractère impératif. Le fournisseur investit spécifiquement pour respecter les délais ; il peut jouer sur cet "actif spécifique" en menaçant de suspendre la livraison, profitant du fait que, compte tenu de l'urgence, son remplacement est difficile (Masten, Meehan et Snyder, 1991).

L'auteur a développé ce concept dans ses travaux sur l'intégration verticale, en étudiant les situations où le choix est limité à un petit nombre de fournisseurs sur un marché. La spécificité des actifs provoque les situations de petit nombre déjà identifiées : même dans les cas où le marché est au départ très ouvert, la mise en oeuvre par les deux parties, au cours de la transaction, d'investissements difficilement redéployables transforme la relation de marché en une relation bilatérale où le fournisseur s'impose de fait. Les mécanismes de régulation du marché sont alors inapplicables, les risques liés à l'opportunisme interviennent, la gestion en interne de la transaction devient plus efficiente.

Williamson a cherché à déterminer les "zones" de meilleure efficacité relative des structures de gouvernance en fonction de la spécificité des actifs. Il s'agit alors pour la firme de minimiser la somme des coûts de transaction et des coûts de production. La figure 6.2 présente les coûts comparés du marché et de la hiérarchie et identifie en fonction du degré de spécificité des actifs, la solution la plus efficiente :

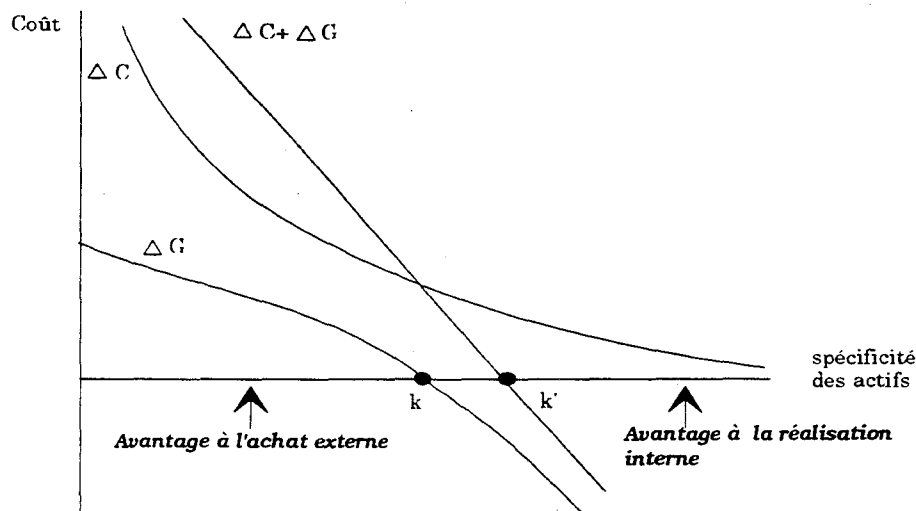


Figure 6.2. Efficacité comparée de la hiérarchie et du marché en fonction du degré de spécificité des actifs (source : Williamson, 1985)

Dans ce schéma,

$$\Delta G = \text{Coûts de transaction de la firme} - \text{coûts de transaction du marché}$$

Les coûts de transaction encourus par la firme sont supérieurs à ceux du marché pour des actifs peu spécifiques ; la situation s'inverse quand la spécificité croît.

$$\Delta C = \text{Coûts de production de la firme} - \text{coûts de production du marché}$$

Pour les biens standardisés, l'avantage de production est au marché qui bénéficie des économies d'échelle et d'étendue. Cet avantage se réduit quand la spécificité augmente.

$$\Delta C + \Delta G \text{ est la résultante des coûts de production et de transaction.}$$

On peut distinguer trois zones :

- en cas de spécificité des actifs faibles (valeur comprise entre 0 et k), la firme a tout intérêt à acheter le bien sur le marché plutôt que de le produire en interne.

- en cas de spécificité des actifs forte (valeur supérieure à k') les gains en coûts de transaction dépassent les pertes en coûts de production. L'entreprise a intérêt à réaliser en interne.

- en cas de spécificité intermédiaire (valeur comprise entre k et k'), l'entreprise doit abriter entre coûts de production et coûts de transaction. Cette zone est également celle où interviendront les formes contractuelles hybrides.

Il est à noter ici que Williamson considère que le coût de production ne peut à lui seul être la justification d'une réalisation interne (ΔC toujours >0). Il s'appuie pour cela sur le principe que les économies d'échelle et d'étendue seront toujours favorables au marché.

2.2 La difficulté d'évaluation du bien

Pour introduire le dernier attribut des transactions, la difficulté d'évaluation du bien ou du service, Williamson fait référence notamment aux travaux d'Alchian et Demsetz (1972) et de Barzel (1982). Cet auteur a établi une synthèse intéressante des problèmes d'évaluation des biens, dont nous reprenons ici les éléments dans le tableau 6.1.

Objectif de l'évaluation	S'assurer que les termes de l'échange sont intéressants pour l'acheteur
Problèmes de mesure	L'attribut intéressant ne peut pas être évalué directement (exemple : goût du jus d'une orange) L'attribut est estimé à l'aide d'une mesure substitutive incomplète (exemple : poids ou aspect du fruit) L'évaluation est reportée dans le futur
Démarche de l'acheteur (deux étapes)	① Sélectionner le fournisseur ② Sélectionner le produit
Moyens de limiter le coût de la mesure	
Sélection du fournisseur	Marque, réputation : pour éviter le coût d'évaluation de chaque produit, le fournisseur engage sa réputation sur la qualité et l'uniformité de ses prestations
Sélection du produit	Garantie sur le produit : la garantie permet de reporter sur l'acheteur, au moment de l'acquisition, le coût de la mesure. Le producteur n'a plus besoin de mettre en place un système de contrôle <i>a priori</i> pour découvrir les défauts. Ils lui sont révélés par le client qu'il satisfait en échangeant le produit
Moyens d'éviter la mesure	
<i>Share contracts</i>	L'incapacité à estimer le résultat <i>a priori</i> conduit à conclure un accord sur la répartition du résultat. Ex : droits d'auteurs.
Suppression de l'information ⁶	Suppression de toute information permettant de discriminer entre les produits (labels d'uniformité) et/ou les fournisseurs (restrictions sur la publicité). Ex : produits présentés sous emballage, profession médicale

Tableau 6.1. L'analyse de Barzel (1982) des problèmes d'évaluation des biens

Nous retrouvons dans cette caractéristique des transactions certains thèmes de la théorie de l'agence qui sont l'incertitude sur la qualité, la difficulté de mesurer le résultat, le rôle des garanties et les effets de la réputation.

⁶ La suppression de l'information sur le produit ne peut s'envisager sans une bonne réputation du fournisseur, qui permet au client d'établir une confiance *a priori* dans sa production ; de même, la suppression de l'information sur le fournisseur ne peut s'envisager sans l'existence d'un organisme régulateur garantissant une qualité minimale et l'homogénéité des prestations.

Dans le cas des biens dont la qualité est difficile à mesurer lors de l'établissement du contrat, l'évaluation du bien et la mise en oeuvre de moyens pour limiter le coût de la mesure ou pour l'éviter représentent des coûts de transaction élevés. Barzel développe l'argument selon lequel l'intégration verticale est un moyen d'éviter l'évaluation du bien à chaque stade intermédiaire de la production, et de remplacer la mesure coûteuse des outputs par la mesure plus facile des *inputs* (le comportement) pour avoir une estimation suffisante du résultat.

III. Les caractéristiques des structures de gouvernance

Alors que la nature des coûts de transaction et les attributs des transactions ont été définis dès les premiers travaux de la TCE, ce n'est que plus tardivement qu'ont été regroupées les dimensions qui différencient les structures de gouvernance. Williamson identifie trois facteurs discriminants qui expliquent les différences de coûts et les capacités de chaque mode contractuel : les capacités d'adaptation, le régime d'administration des contrats (*contract law*) et le mode de pilotage de la transaction, c'est-à-dire le système d'incitations et de contrôle.

3.1 Les capacités d'adaptation

Dans son article Comparative Economic Organization: The Analysis of Discrete Structural Alternatives (1991a), Williamson s'appuie sur les travaux de Hayek (1945) et Barnard (1938) pour comparer deux formes d'adaptation :

1. Pour Hayek, l'adaptation au changement est le problème central de l'économie : "*the economic problem of society is mainly one of the rapid adaptation in the particular circumstances of time and place*". Le **système de prix**, bien plus qu'une planification centralisée, est le mécanisme efficient pour communiquer l'information et provoquer l'adaptation.

2. Pour Barnard, l'adaptation est également le problème central de l'organisation. Mais c'est l'**organisation interne** cette fois-ci qui est le mécanisme privilégié d'adaptation⁷.

Williamson explique cette divergence par une approche contingente de l'adaptation, dont il distingue ainsi deux formes. L'une, évoquée par Hayek, est dite de type "A" où le A réfère à l'Autonomie des agents sur le marché. C'est par cette autonomie que se fait l'adaptation, au travers du mécanisme des prix, chaque agent cherchant individuellement à maximiser son utilité. L'autre type d'adaptation, mise en évidence par Barnard, est une adaptation de type "C" (Coopérative), résultant d'une coopération consciente, délibérée, volontaire.

Chaque forme d'adaptation est efficiente dans un **contexte donné** (tableau 6.2).

	Conditions d'application
Adaptation Autonome (Hayek) : Marché	<ul style="list-style-type: none"> • Le prix est le vecteur de toute l'information commune • Les agents agissent individuellement, chacun pour maximiser son utilité • Limites : toutes les situations où le système de prix ne joue pas efficacement son rôle et où la concurrence ne peut pas pleinement s'exercer⁸
Adaptation Coopérative (Barnard) : Organisation interne	<ul style="list-style-type: none"> • L'objectif commun est l'information partagée • L'individu choisit d'entrer dans un système coopératif en fonction des satisfactions potentielles offertes par le système (<i>incentives</i>) • Limites : capacité de l'organisation à fournir des <i>incentives</i> efficaces, coût d'organisation de la coopération

Tableau 6.2. Conditions d'efficience des deux formes d'adaptation (d'après Williamson, 1991a)

On voit que ce contexte dépend fortement des attributs des transactions :

- le prix est le vecteur de toute l'information sur des marchés banalisés (actifs non spécifiques et produits ne présentant pas de problèmes d'évaluation du bien *a priori*),

⁷. Ce débat a été parfaitement illustré par A. Chandler, qui a opposé la "main visible" des managers (1977) à la "main invisible du marché" évoquée par A. Smith.

⁸. L'adaptation autonome ne peut donc pas jouer en cas de relation à long terme engageant des actifs spécifiques, puisque dans ces situations la libre concurrence initiale se transforme en relation de petit nombre.

- l'adaptation coopérative est nécessaire en cas d'incertitude forte ainsi que dans toutes les situations d'insuffisance du marché (actifs spécifiques, difficulté d'évaluation du bien).

3.2 Le régime d'administration des contrats (*contract law*)

Williamson associe (1991a) une forme contractuelle privilégiée à chaque structure de gouvernance : le contrat classique pour le marché, le contrat néoclassique pour les formes hybrides et la doctrine de l'indulgence (*forbearance*) pour l'organisation interne⁹.

L'auteur développe ici sa vision de la firme : en particulier, il considère que **la spécificité de l'organisation interne en tant que forme contractuelle est de s'appuyer sur un système d'obligations "élastiques" laissant une large part à l'adaptation et au pardon des erreurs et des fautes**. Williamson en veut pour preuve le fait que les tribunaux ne jugent pas les conflits entre services d'une même firme concernant la fixation de prix de cession interne, le non respect des délais ou la mauvaise qualité des biens ou services. De même, les conflits entre le travailleur et l'entreprise sont résolus en priorité en interne. L'organisation interne est donc "sa propre cour d'appel". Celle-ci fonctionne davantage sur un système global d'autorité que sur l'application stricte de règles contractuelles.

La contrainte contractuelle, même exprimée sous forme de mécanismes visant à recréer des relations de marché entre les différents acteurs de l'entreprise (prix de cession interne, relations "clients-fournisseurs"), ne sera donc pas comparable à celle s'exerçant sur le marché. En contrepartie, la définition par la hiérarchie d'objectifs communs favorise l'adaptation séquentielle tout en évitant les coûts de renégociation ; les systèmes de contrôle permettent de définir des règles et de pénaliser les comportements opportunistes. Ces comportements sont moins susceptibles de se produire car l'organisation interne permet le développement d'une culture commune et d'une relation de confiance s'inscrivant dans la durée.

⁹. Dans un précédent ouvrage (1985) l'auteur avait proposé un tableau de correspondance légèrement différent, à partir de trois formes contractuelles définies par MacNeil (1974) - les contrats classiques, néoclassiques et personnalisés.

Dans certains cas, plutôt que de tenter de simuler les règles du marché en interne, il est plus efficient pour Williamson d'utiliser des structures de gouvernance hybrides alliant un régime contractuel plus contraignant à des structures de pilotage personnalisées.

Nous utilisons les éléments présentés dans cet article pour composer le tableau 6.3.

	Régime d'administration des contrats		
	Contrat classique	Contrat néoclassique	Doctrine de l'indulgence
Identité des parties	anonyme, indépendance	personnalisé, autonome dépendance bilatérale	personnalisé, forte dépendance
Champ contractuel			
Etendue du contrat	limitée et précise	transaction limitée mais cadre ajustable	accord global de coopération
Durée	immédiat	long terme	long terme
Complétude	complet	incomplet	incomplet
Obligations			
Traitement des litiges	pré-déterminé	renégociation	adaptatif, séquentiel
Respect des obligations	strict	tolérance (<i>excuse</i>)	indulgence (<i>forbearance</i>)
Juridiction	tribunal	arbitrage	autorité (<i>fiat</i>)
Système de pilotage	régulation par le prix, règles formelles	mixte (marché et structure de pilotage spécifique)	motivation par la hiérarchie, systèmes de contrôle
Mode d'organisation associée	Marché (Exemple : vente)	Hybride (Exemple : contrats de franchise)	Hiérarchie (Organisation interne)

Tableau 6.3. Correspondance entre structures de gouvernance et régimes d'administration des contrats (d'après Williamson, 1991a)

3.3 Le système d'incitations et de contrôle

Outre les capacités d'adaptation et le régime d'administration des contrats, les structures de gouvernance se distinguent également par les systèmes de motivation et de contrôle qu'elles peuvent mettre en place. Williamson reprend ici les deux types de rémunération identifiés par la

théorie de l'agence, celle liée au résultat et celle liée au comportement, mais il attribue respectivement ces deux formes au marché et à la hiérarchie :

1. Le marché dispose de forts facteurs de motivation (*high-powered incentives*) car il lie directement la rétribution au résultat obtenu. Chacun travaille directement à l'accomplissement de ses propres objectifs en cherchant à maximiser sa propre utilité. En revanche, il ne dispose pas de systèmes de contrôle du comportement, ceux-ci n'étant pas efficaces, comme nous avons pu le voir avec la théorie de l'agence, dans le cas de contrats basés sur le résultat.

2. Comparée au marché, l'organisation interne ne propose selon Williamson que des facteurs de motivation faibles (*flat or low-powered incentives*). Cet argument, développé dans l'ouvrage de 1985 (chapitre 6) s'appuie sur le fait que la firme ne joue jamais "à fond" le jeu du marché : en cas d'échec, la pénalité encourue par un service est négociable et en cas de très grand succès, la firme n'est pas prête à "payer le prix", c'est-à-dire à rémunérer complètement en fonction du résultat. L'existence de *high incentives* personnalisés serait de plus contraire à l'établissement du climat de coopération, car elle renforcerait les comportements individualistes des acteurs. La firme va donc mettre en place des systèmes administratifs de contrôle qui vont permettre de connaître le comportement des agents. Ces systèmes, bien que coûteux, sont efficaces quand la relation s'inscrit dans une perspective à long terme.

3.4 Synthèse des caractéristiques des structures de gouvernance ; précision du lien avec la spécificité des actifs.

Williamson dresse ainsi un "portrait-type" des différentes structures de gouvernance. Les caractéristiques attribuées à chaque structure se renforcent pour former un ensemble cohérent :

- les échanges limités et précis du marché permettent une évaluation immédiate du résultat et la mise en oeuvre de récompenses (rémunération) et sanctions (respect strict des obligations). La sélection par le prix provoque l'adaptation.

- l'adaptation coopérative interne n'est envisageable que dans le cadre d'un contrat personnalisé où la recherche de l'intérêt personnel n'est pas encouragé par une rémunération liée directement au résultat. Le contrôle des comportements est continu et s'exerce de façon formelle et informelle ; la coopération est volontaire et globale (participer aux objectifs de l'entreprise).

Attributs*	Structures de gouvernance		
	Marché	Formes Intermédiaires	Hiérarchie
Pilotage de la transaction :			
Intensité des facteurs de motivation	++	+	0
Contrôles administratifs	0	+	++
Capacités d'adaptation :			
Adaptation autonome	++	+	0
Adaptation coopérative	0	+	++
Contraintes associées au Système contractuel :			
(<i>Contract law</i>)	++ (classique)	+ (<i>excuse</i>)	0 (<i>forbearance</i>)
* ++ = forme forte, + = forme semi-forte, 0 = forme faible			

Tableau 6.4. Caractéristiques des structures de gouvernance (source : Williamson, 1991a)

Section 2 - L'application au logiciel

La théorie de l'agence nous avait permis de classer les formes contractuelles en fonction du mode de pilotage (le développement interne et la régie correspondant à une évaluation du comportement, le forfait correspondant à une évaluation par le résultat). Nous pouvons maintenant associer ces trois formes aux structures de gouvernance de Williamson :

- l'informatique interne correspond à la hiérarchie,
- la régie allie des contrats personnalisés entre deux parties autonomes, une relation souvent à moyen ou long terme mais aussi une structure de pilotage et des systèmes de contrôle qui sont ceux de l'informatique interne ; on est donc en présence d'un mode d'organisation de forme hybride,
- le forfait, pour lequel l'évaluation se fait sur le résultat produit et où le fournisseur assure la gestion du projet ; le forfait se rapproche d'un mode d'organisation de type marché.

Williamson dote chacune de ses structures de gouvernance de caractéristiques fortes. Ces caractéristiques visent à décrire des idéaux-types (marché, hiérarchie, formes hybrides) ; encore faut-il que ces idéaux-types se traduisent concrètement dans les situations réelles étudiées. En particulier, **il considère que l'organisation interne (la hiérarchie) est le lieu privilégié de l'adaptation coopérative.**

La coopération au sein des projets est un aspect essentiel du développement d'applications mais, contrairement aux hypothèses du modèle de Williamson, beaucoup de chercheurs du domaine des systèmes d'information ont insisté sur les lacunes de la coopération entre utilisateurs et informaticiens internes. Nous reviendrons donc dans un premier temps sur ce point afin de voir dans quelle mesure l'argument d'une bonne adaptation coopérative interne est pertinent pour notre domaine.

Le second élément sur lequel nous souhaitons revenir est lié à la constatation pratique de la coexistence pour la même activité des trois grandes structures de gouvernance identifiées par Williamson (développement interne, régie, forfait). Suivant la théorie des coûts de transaction, l'utilisation différenciée de ces trois modes devrait s'expliquer par des caractéristiques

différentes des transactions. Nous allons revenir sur les attributs des projets et en particulier préciser dans le cadre de l'activité de développement les notions de spécificité des actifs engagés et de difficulté d'évaluation du bien.

I - Le développement en interne permet-il une bonne adaptation coopérative?

L'environnement humain et organisationnel joue un rôle considérable dans l'ensemble des étapes du cycle de vie du logiciel. En particulier, et pour nous limiter à la question posée ici, de nombreux auteurs ont souligné le poids de la coopération entre utilisateurs et informaticiens dans le processus de développement. Ce thème a donné lieu à une grande variété d'approches s'organisant autour de la relation avec les utilisateurs, vue au niveau individuel et psychologique avec les concepts de styles cognitifs (Mintzberg, 1976, Robey et Taggart, 1981), de nature de la tâche (Ives et Olson, 1981, Glass et al., 1992), d'implication des utilisateurs (*user involvement*) (Swanson, 1974, Ives et Olson, 1984, Franz et Robey, 1986, Barki et Hartwick, 1989, Jarvenpaa et Ives, 1991), de rôle des informaticiens (Applegate et Elam, 1992, Stephens et al., 1992), de communication entre informaticiens et utilisateurs (Feeny et al., 1992). Ces différents thèmes sont pertinents pour notre recherche. Nous allons tenter d'en faire la synthèse en nous interrogeant tout d'abord sur les facteurs généraux pouvant limiter la coopération entre utilisateurs et informaticiens ; nous ferons ensuite le point sur les études analysant la relation entre utilisateurs et informaticiens internes ; nous terminerons par l'analyse de la coopération au sein des projets de développement.

1.1 Les facteurs généraux susceptibles de limiter la coopération entre informaticiens et utilisateurs

Robey et Markus (1984) identifient quatre classes d'opposition entre développeurs et utilisateurs, sur lesquelles nous allons revenir en détail :

- le style cognitif et la personnalité,
- les caractéristiques des tâches effectuées et les critères d'évaluation associés,

- le cadre de référence et le système de loyauté,
- la situation dans la structure organisationnelle.

a) le style cognitif et la personnalité

Certains chercheurs en management ont cherché à intégrer les travaux des neurologues et psychologues dans leur réflexion. H. Mintzberg a ainsi insisté sur la double dimension du management (*Planning on the left side and managing on the right*, 1976), correspondant aux activités respectives des hémisphères cérébraux gauche - logique, rationnel, analytique - et droit - créatif, intuitif, spatial et heuristique.

Plusieurs modèles sur les styles cognitifs ont été proposés dans la littérature¹⁰. En particulier on a cherché à savoir si des différences individuelles de style cognitif pouvaient conduire à privilégier certains types de systèmes d'information. Nutt (1979) identifie quatre styles de décideurs (sensitif-rationnel, sensitif-affectif, intuitif-rationnel, intuitif-affectif) et propose une typologie de systèmes d'aide à la décision adaptés à chaque style cognitif.

On a pu voir ainsi dans l'informatique et les informaticiens le triomphe du mode "sensitif-rationnel" (recherche de données, valeurs d'objectivité, esprit d'analyse, rationalité, formalisme), ce qui rend difficile la communication avec les utilisateurs privilégiant d'autres modes : "les systèmes d'information informatisés sont souvent conçus par des individus qui perçoivent le processus décisionnel comme étant très systématique. Les gestionnaires de style systématique sont généralement satisfaits par cette approche. Les gestionnaires intuitifs sont par contre plus réticents, car les spécifications ne sont pas conformes à leur style cognitif." (Davis et al., 1986, vol. 2, p. 70). Une recherche précédente (Delmond, 1990, p.53) nous a fourni une illustration de ce problème de communication et des solutions envisageables pour le surmonter : un responsable d'informatique déclare ainsi avoir modifié ses méthodes de travail traditionnelles (cycle en cascade avec une phase d'analyse et de description très formalisée) lors

¹⁰. On trouvera une très bonne introduction ainsi qu'une intéressante revue de la littérature sur ces concepts dans le chapitre 3, volume 2 de Systèmes d'information pour le management (Davis, Olson, Ajenstat et Peaucelle, 1986).

du développement d'un système d'information pour le service marketing, simplement parce que "les gens du marketing ont une approche artistique, intangible, inquantifiable; nous devons utiliser des techniques nous permettant de dialoguer, comme le maquettage".

Le succès de la micro-informatique et de l'interface graphique est en grande partie dû à la réhabilitation du cerveau droit (visuel, heuristique) dans la relation homme/machine. Les nouvelles technologies de développement étudiées dans le chapitre 3, utilisant des techniques de prototypage, favorisent le dialogue utilisateur-informaticien. Le critère technique du mode de développement utilisé apparaît ainsi comme un facteur majeur favorisant la coopération : si l'informatique interne se limite aux modes de développement classiques (cycle en cascade), certains utilisateurs peuvent être amené à rechercher à l'extérieur des modes de développement plus proches de leur style cognitif.

b) les caractéristiques des tâches effectuées et les critères d'évaluation associés

Le travail du développeur doit-il être rapproché de celui du chercheur ou du mécanicien ? Identifier la nature de la tâche permet en effet de déterminer le "profil idéal" de celui qui l'exécute, et donne en même temps des indications sur le mode d'évaluation approprié. De plus, l'image de la fonction et celle de celui qui l'occupe se confondent pour former une **identité globale perçue par les autres groupes de l'organisation.**

Glass, Vessey et Conger (1992) ont cherché à mesurer la nature des tâches dans la première étape du développement d'une application, en appliquant deux protocoles de recherche différents :

- dans un premier temps, la recherche s'appuie sur une *évaluation par deux experts ayant une expérience de plusieurs années dans la fonction.* Partant d'une taxonomie de tâches élaborée après une recherche dans la littérature, les auteurs ont demandé à ces experts de classer les tâches dans trois catégories, intellectuelle, technique (*clerical*) et indéterminée. Les résultats sont convergents avec ceux obtenus lors d'études précédentes et montrent une dominance nette de tâches intellectuelles.

- dans un deuxième temps, les auteurs ont mesuré les temps passés sur chaque type de tâche dans une expérimentation conduite sur six sujets en laboratoire¹¹. Les conclusions sont convergentes avec l'étude précédente.

Les auteurs concluent que le développement d'applications est une tâche de nature principalement intellectuelle et regrettent l'écart significatif entre la réalité de la tâche et la perception plus restrictive qu'en a le management.

Le reproche méthodologique que l'on peut adresser aux deux auteurs est évident : le jugement exercé par les deux experts peut ainsi être motivé par leur appartenance au groupe des informaticiens ; il ne leur est pas possible d'émettre une opinion dévalorisante de la tâche sans provoquer un phénomène de dissonance cognitive.

Mais surtout cette recherche néglige à notre avis un aspect essentiel du débat : l'opinion sur l'activité de développement ne ressort pas tant d'un jugement de **fait** sur la tâche que d'un jugement de **valeur** sur la fonction. Deux grandes approches s'affrontent, correspondant respectivement à une vision "intellectuelle" et une vision "technicienne" de l'activité des développeurs, que l'on peut opposer point par point, comme nous l'avons fait dans le tableau 6.5.

¹¹. Les sujets sont des étudiants novices en systèmes d'information suivant un cours de conduite de projet dans une *business school*. On mesure le temps passé à réfléchir (tâche intellectuelle) et celui passé à écrire (tâche mécanique). L'étude de Glass et al. comporte cependant un ensemble de limites graves qui remettent en cause la validité des résultats obtenus, ce malgré un appareillage de mesure très élaboré : population non représentative de la population effectuant normalement la tâche, durée d'expérimentation de deux heures, opérationnalisation des concepts discutables. Enfin il faut préciser que seules les tâches de l'étape de spécification ont été étudiées (et donc pas par exemple celles de l'étape de réalisation).

Vision "intellectuelle"	Vision "technicienne"
<i>Caractéristiques de l'activité de développement :</i>	
intellectuelle créative hautement spécialisée et qualifiée complexe artisanale par nature	routinière, mécanique, automatisable très codifiée, normalisée ne requérant qu'une compétence technique facile industrielle
<i>Facteur déterminant dans le processus :</i>	
l'homme (expertise, créativité)	l'ensemble des outils (<i>Case</i> , etc..)
<i>Evaluation et mode de pilotage appropriés :</i>	
aléa irréductibles rôle dominant de l'expert (informaticien)	production et résultats contrôlables rôle dominant du généraliste (<i>manager</i>)

Tableau 6.5. Les deux visions des tâches de développement

Pour notre part, au-delà d'une discussion sur la nature "objective" de la tâche, il nous semble important de souligner **les divergences de perception** entre ceux qui l'exercent et une partie des autres acteurs de l'organisation, utilisateurs de l'informatique et/ou décideurs. Ces divergences de perception ne s'arrêtent d'ailleurs pas, tant s'en faut, à cet aspect; on peut en fait parler plus globalement de "cadre de référence" différent.

c) le cadre de référence et le système de loyauté.

L'informaticien est dans l'organisation un professionnel, c'est-à-dire un opérateur "qui fait un travail exigeant un degré élevé de savoir-faire et de connaissances" (Mintzberg, 1986, p. 198). La spécificité de la tâche de l'expert lui confère certains avantages : "Non seulement le travail devient attirant et intéressant pour les opérateurs, mais sa complexité empêche tout contrôle administratif rigide" (Mintzberg, 1986, p. 198).

Les spécificités des professionnels dans l'organisation ont été depuis longtemps identifiées par les sociologues. Selon Perrow (1961), "les professionnels ont des intérêts qui leur sont propres, intérêts qui donnent la forme de l'organisation. Ils peuvent développer une identité et une éthique qui les coupent de la communauté et qui privilégient des buts spécialisés, étroits et aux yeux de la critique bénéfiques pour eux-mêmes".

Mintzberg (p. 199 et s.) analyse les buts des professionnels, qui s'expriment en tant que groupe "les buts du groupe incluent la protection du groupe et le maintien de leur autonomie, la tentative de rehausser le prestige du groupe et d'accroître leur force par le nombre d'adhérents et l'augmentation des ressources" et également au niveau individuel "l'un des buts est souvent la poursuite de la perfection professionnelle, parce que l'opérateur professionnel a tendance à aimer son travail et donc il s'engage plus fortement dans son travail". Le professionnel place la perfection dans son art avant le respect des objectifs généraux de l'organisation, ce qui constitue une situation potentiellement génératrice de conflits et peut rendre sa loyauté *a priori* suspecte aux yeux de l'organisation, car, comme l'indique Mintzberg, "naturellement la poursuite de la perfection professionnelle peut se faire parfois en dehors des besoins de l'organisation ou bien de ses clients, comme le chirurgien par exemple qui aurait tendance uniquement à opérer pour améliorer son habileté au bistouri". A combien de responsables du développement d'applications n'a-t-on pas reproché d'être des chirurgiens du logiciel ?

L'informaticien peut ainsi être perçu par les non-informaticiens à la fois comme "celui que l'on ne comprend pas", avec lequel la communication est perturbée par un langage ésotérique, "celui que l'on ne peut pas contrôler", parce qu'il est le seul détenteur des secrets de l'organisation de sa tâche, "celui qui s'isole", - la séparation géographique des équipes de réalisation et d'exploitation informatique du reste de l'organisation en est une manifestation - et enfin "le traître en puissance", parce que ses centres d'intérêt - dont on ne perçoit que la dimension technologique (Lesca, 1990) -, son pouvoir et son affiliation, plus professionnelle

qu'organisationnelle, lui donneraient à la fois les motifs et l'opportunité de mener un jeu politique au service de ses propres intérêts¹².

d) la situation dans la structure organisationnelle

Quand on parle de la situation des informaticiens dans les organisations, deux points de vue contradictoires s'expriment souvent au sein du *même* discours : d'une part, on dira que l'informatique est un "état dans l'état", lui attribuant ainsi un pouvoir exorbitant, d'autre part on parlera de l'informaticien comme "d'un homme relégué, marginalisé, ne bénéficiant pas des mêmes promotions que les responsables des autres fonctions, n'ayant pas d'image nette et de véritable identité intégrative dans l'entreprise" (Lesca, 1990). On retrouve ici le phénomène de "caste" mis en évidence par Morley (1991).

1.2 La communication entre les informaticiens et les autres membres de l'organisation

Quelle est la qualité de la communication entre informaticiens et utilisateurs¹³? Il y a une dizaine d'années, le thème de la communication entre informaticiens était relativement peu exploré. Les recherches menées sur le terrain à cette époque rejoignent le sentiment général d'un manque de communication entre informaticiens et utilisateurs au sein de l'entreprise. Depuis, de nombreuses études ont été menées, en particulier concernant les responsables de systèmes d'information, qui semblent indiquer une amélioration quantitative et qualitative des contacts dans les toutes dernières années.

¹². Le taux de rotation très élevé des informaticiens dans les années 80 reflète cette fidélité au métier davantage qu'à l'organisation. Pour Nolan (1976), le problème n'est d'ailleurs pas tant la conséquence de la croissance rapide de cette activité que d'une inaptitude de la part des informaticiens et des organisations à définir durablement un mode de coopération harmonieux.

¹³. Nous employons le vocable "utilisateur" bien qu'il nous gêne quelque peu car il pourrait faire penser que l'on présuppose la nature du lien entre les informaticiens et les autres membres de l'organisation. Nous verrons que certains de nos interlocuteurs dans la recherche terrain ont symboliquement préféré abandonner ce terme pour le remplacer par "client".

En 1981, Ives et Olson, reprenant la méthodologie d'observation structurée¹⁴ popularisée dans le domaine de la gestion des organisations par Mintzberg (1973), notent que "les contacts entre les responsables de système d'information et les responsables fonctionnels du même niveau sont remarquablement absents". Les résultats indiquent que les responsables des systèmes d'information consacraient alors seulement 16% de leur temps à des contacts avec des responsables d'autres services contre trois quarts de leur temps à leurs subordonnés.

Une étude plus récente de Stephens et al. (1992) semble indiquer une forte évolution, avec 58% du temps des responsables des systèmes d'information (*Chief Information Officer*) consacré à des contacts en dehors de leur département. Cependant, le faible nombre de cas observés (5) ne permet pas de généraliser ces résultats. Ceux-ci sont confirmés par une recherche de L. Applegate et J. Elam (1992) portant sur les responsables des systèmes d'information de 81 entreprises de secteurs variés, interrogés sur l'occupation de leur temps (figure 6.3.).

a) temps d'interaction avec les membres de trois groupes différents : leur propre service informatique, les directions opérationnelles, les services centraux du groupe (premier tableau)

b) temps consacré à différentes activités du management des systèmes d'information (second tableau).

On a distingué dans l'enquête les cadres ayant pris récemment leur fonctions de ceux les assurant depuis plusieurs années, ce afin de voir s'il existe une différence entre "le responsable de systèmes d'information traditionnel" et "les nouveaux responsables de systèmes d'information". A tous on a demandé quelle était leur expérience professionnelle antérieure (troisième tableau).

¹⁴. Le chercheur accompagne le sujet étudié dans ses activités quotidiennes et mesure les temps consacrés à différents types de tâches. Cette méthode avait permis à Mintzberg d'infirmer un ensemble d'*a priori* largement répandus sur les managers.

L'activité du *Chief Information Officer* : interactions (1)

% de leur temps passé en interaction avec les membres des trois groupes suivants :	Nouveaux responsables	Responsables confirmés
service informatique	41%	40%
directions opérationnelles	31%	40%
services groupe	28%	20%

L'activité du *Chief Information Officer* : tâches (2)

% du temps consacré à chaque activité	Nouveaux responsables	Responsables confirmés
Planification stratégique de l'informatique	27%	16%
Architecture informatique	19%	20%
Gestion des ressources humaines	17%	20%
Exploitation	13%	17%
Développement et maintenance des systèmes	9%	14%
Gestion du risque informatique	6%	3%
Autres activités (non informatique)	7%	-
Autres activités (informatique)	2%	10%

L'expérience professionnelle antérieure (3)

Expérience acquise d'au moins cinq ans dans une fonction..	Nouveaux responsables	Responsables confirmés
..technique (informatique)	19%	60%
..managériale (autres fonctions)	33%	28%
Expérience Mixte (+ de 5 ans dans les deux types de fonctions)	37%	12%

Figure 6.3. Profil et occupations des responsables de systèmes d'information : principaux résultats de l'étude de Applegate et Elam (1992)

Les résultats de cette étude confirment **l'évolution du rôle du responsable des systèmes d'information** : plus orienté vers la stratégie et moins vers les tâches techniques, possédant une expérience managériale et créant des réseaux d'interaction à tout niveau (groupe et divisions opérationnelles). Les profils des nouveaux arrivants dans la fonction et celui des responsables traditionnels sont remarquablement différents et beaucoup plus "managérier". L'évolution la plus remarquable est sans doute celle du pourcentage de *CIO* pouvant revendiquer le titre d'expert (plus de 5 ans dans une fonction informatique technique), qui chute de 60% à seulement **19%** chez les nouveaux responsables.

Feeny et al. (1992), s'appuyant sur une série d'entretiens en profondeur auprès des *CEO* (*Chief Executive Officer*) et *CIO* de 14 entreprises anglaises, ont établi les caractéristiques idéales du responsable de systèmes d'information, vu par les *CEO* :

- honnêteté, intégrité, sincérité, ouverture,
- orientation managériale, motivation, capacité à communiquer dans un langage clair,
- homme de communication, *leader*, politique, capable de construire un réseau de relations et de vendre les systèmes d'information en interne,
- connaissant les évolutions techniques et sachant déterminer leur impact stratégique pour l'entreprise,
- homme d'équipe, promoteur et catalyseur du changement, porteur d'une vision à long terme différente.

Le rapprochement global entre les informaticiens, d'une part, et les membres d'autres services de l'entreprise ou de la direction générale, d'autre part, constitue un enjeu majeur de la prochaine décennie pour les responsables de fonctions informatiques (Selig, 1991, Lesca, 1990). Ce rapprochement global semble nettement progresser dans les dernières années. Il nous reste maintenant à voir ce qu'il en est de la coopération au sein des projets de développement.

1.3 La coopération avec les utilisateurs au sein des projets de développement

L'ensemble des manuels sur les systèmes d'information s'accorde pour voir dans la participation des utilisateurs une condition *sine qua non* du succès d'un développement spécifique. Les risques d'une mauvaise coopération sont un mauvais arbitrage entre les projets, l'inadéquation fonctionnelle, l'abandon de projets, le dépassement des budgets, le rejet ou la sous-utilisation des applications.

Quelle forme doit prendre cette coopération ? Deux approches ont successivement été adoptées pour tenter de décrire et de prescrire la nature des interactions entre informaticiens et utilisateurs. Les premières recherches effectuées se focalisent sur l'utilisateur final, les étapes du processus, les tâches effectuées et l'acceptation d'un nouveau système. Nous serons guidés dans leur interprétation par l'excellente revue de la littérature faite par Ives et Olson en 1984. Le second groupe de recherches, plus récent, s'intéresse davantage à l'implication du management, l'attitude psychologique, favorable ou défavorable envers les systèmes d'information, et la politique générale de l'entreprise envers les technologies de l'information (comportement de précurseur ou de "suiveur").

1.3.1. les premières recherches : la participation opérationnelle au processus de développement

Quand l'utilisateur doit-il intervenir ? Quel type de participation peut-il adopter ? Quel est son degré d'engagement dans le processus ? L'hypothèse des nombreuses recherches menées dans les années 70 est qu'une plus grande participation des utilisateurs pendant le processus de développement facilite l'adhésion au nouveau système.

Ives et Olson (1984) identifient deux fondements théoriques à cette hypothèse : le premier est le concept de management participatif (*participative decision-making*), selon lequel la possibilité pour un subordonné de participer aux décisions de management concernant son propre travail améliore la productivité et la satisfaction au travail. Le deuxième est la théorie de la gestion du changement organisationnel (*planned organisational change*) qui définit le changement comme le résultat d'un effort commun ou d'une négociation.

L'étude de Swanson (1974) est représentative de cette tendance. L'auteur a cherché à poser et à prouver les liens entre la participation des utilisateurs en cours de processus (*a priori involvement*), l'utilisation des systèmes d'information (*inquiry involvement*) et le niveau de satisfaction. Selon Swanson, la plus grande implication des utilisateurs entraîne un niveau de satisfaction plus élevé, qui conduit à une utilisation plus intensive du système, celle-ci renforçant à son tour la satisfaction (figure 6.4.).

Premiers modèles : participation de l'utilisateur opérationnel au processus de développement : relations entre le degré de participation au processus, le niveau de satisfaction et l'utilisation du système.

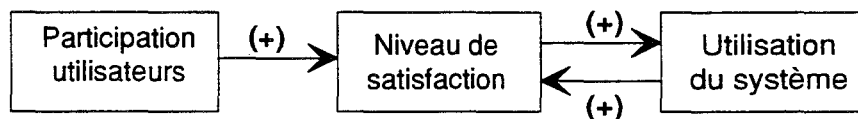


Figure 6.4. Premiers modèles de participation des utilisateurs, d'après Swanson, 1974 (résultats contestés par Ives et Olson, 1984)

Les règles de l'art vont donc établir la participation des utilisateurs selon les caractéristiques suivantes :

- la participation est celle de l'utilisateur final,
- celui-ci mène un ensemble d'interventions et de contrôles pendant le processus de développement : dans un cycle classique en cascade, l'utilisateur final -ou son représentant- doit au minimum définir ses besoins (phase d'analyse préalable et de conception générale), valider l'application dans les phases de tests et participer à la phase de lancement en effectuant une partie de la formation dans les sites d'utilisation. Cependant, on met aussi en garde contre une trop grande implication des utilisateurs en phase de réalisation qui risque de conduire à une remise en cause des choix initiaux et à une instabilité dommageable au succès du projet. La participation opérationnelle des utilisateurs est donc recommandée en début (design) et en fin (tests, mise en oeuvre) de processus,

- le style de participation peut être plus ou moins direct : consultatif, représentatif ou consensuel, avec une préférence pour la forme consensuelle,

- la participation est particulièrement nécessaire quand le besoin est mal structuré ou difficile à communiquer (on retrouve ici les travaux déjà évoqués sur le prototypage) ou quand il y a des risques importants de rejet (changement important, fort impact organisationnel),

- en réaction au constat des effets négatifs d'une participation inexistante ou symbolique émerge une idée qui a fait largement son chemin depuis, celle de rendre responsables les utilisateurs en imputant les développements sur leurs propres budgets ou en intégrant ce facteur dans l'évaluation de leur performance. La gestion de projet (suivi des temps, des budgets, des résultats) devient du même coup une préoccupation des utilisateurs.

Cependant, et bien que de nombreux chercheurs cherchent à valider empiriquement ce modèle de participation, les tests n'apportent que résultats mitigés (Ives et Olson, 1984). Cette situation va conduire quelques chercheurs à redéfinir les concepts et à s'intéresser à une autre facette du rôle de l'utilisateur : ce n'est plus l'utilisateur au sens strict du terme, mais le **prescripteur** (direction générale, management), qui va être au centre de ce deuxième groupe d'études.

1.3.2 le second groupe de recherches : l'implication du management, les politiques de leader et de suiveur dans l'utilisation des systèmes d'information

Le concept de "participation des utilisateurs" se diversifie et s'enrichit :

- ainsi on distingue la participation opérationnelle, qui fait référence aux actions (*user participation*), de l'implication (*user involvement*) qui décrit plutôt un état d'esprit, une motivation (Barki et Hartwick, 1989)¹⁵,

¹⁵. Barki et Hartwick définissent l'implication de cette manière : "this article proposes to define user involvement as subjective psychological state. A user is involved when he or she considers a system to be both important and personally relevant". (1989, p. 53)

- le terme "utilisateur" est réservé à l'utilisateur final potentiel de l'application. On étudie également la participation ou l'implication du management (*executive participation* et *executive support*), prescripteur des projets.

Par ailleurs, le résultat observé n'est plus le succès ou l'échec d'un projet donné, mais on cherche à obtenir une mesure globale de la satisfaction et de l'utilisation des systèmes d'information. Jarvenpaa et Ives (1991) demandent par exemple à 83 *CEO* d'évaluer la position de leur entreprise au sein de leur secteur d'activité, en indiquant si elle se classe plutôt du côté des précurseurs dans l'utilisation des systèmes d'information ou si elle adopte un comportement de suiveur. Les auteurs ont testé trois modèles alternatifs en distinguant la participation des dirigeants (critère objectif) de leur implication (critère subjectif reflétant les opinions vis à vis de l'informatique).

Nous reprenons dans la figure 6.5. les résultats de cette étude. Un (+) ou un (-) indique une corrélation positive ou négative ; la qualité du test (niveau de signification) est indiquée au dessous.

Second groupe de modèles de participation des utilisateurs, concernant les dirigeants : relations entre participation, **implication** et comportement de leader de l'entreprise dans l'utilisation des systèmes d'information au sein d'un secteur

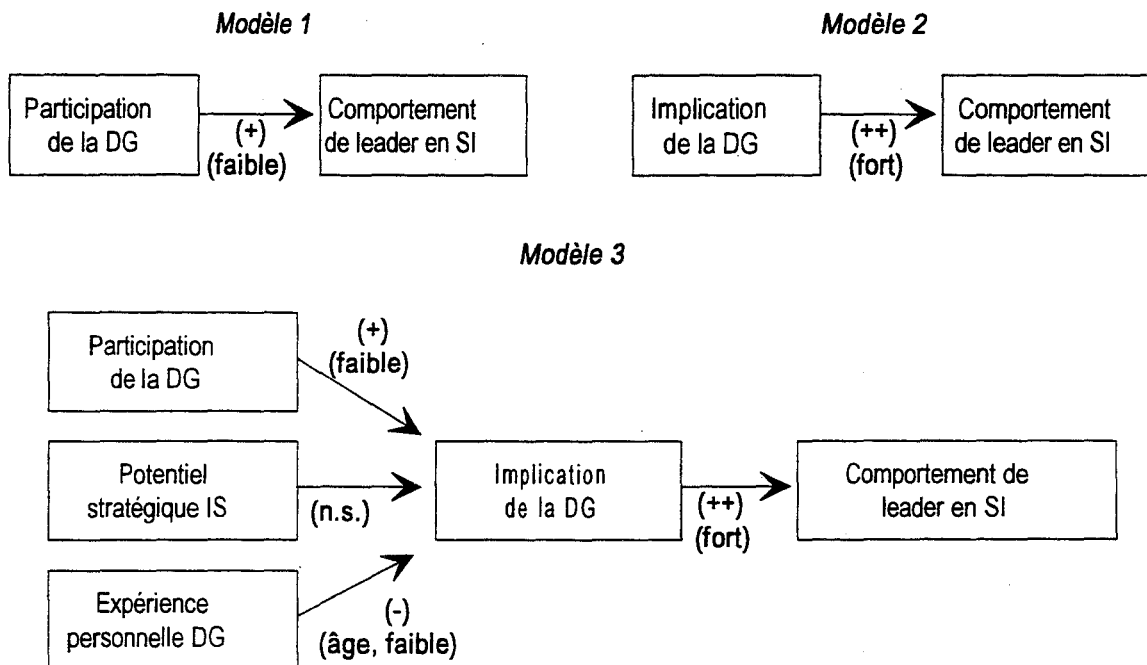


Figure 6.5. Second groupe de modèles de participation des utilisateurs
(d'après Jarvenpaa et Ives, 1991)

Les résultats de cette recherche sont à plusieurs égards en contradiction avec certaines opinions traditionnellement admises dans le champ des systèmes d'information :

- alors que l'on n'a cessé de mettre l'accent dans les manuels sur la participation opérationnelle des utilisateurs au développement, il semble que **l'opinion, l'état d'esprit de leurs dirigeants vis-à-vis de l'informatique** soit un facteur plus déterminant mais cependant mal cerné et moins reconnu.

- l'opinion des dirigeants semble avoir **peu de relations avec une utilisation personnelle** de l'informatique (fort peu répandue à ce niveau hiérarchique) **ni même avec une implication directe dans le management de l'informatique**. Selon Feeny et al. (1992), l'opinion peut en revanche être liée à l'expérience personnelle de projets réussis et influencée par la participation à des séminaires sur les systèmes d'information. Jarvenpaa et Ives ont

également trouvé un lien entre l'âge des *CEO* et leur attitude, les plus jeunes s'impliquant plus directement dans le management des systèmes d'information.

En conclusion sur cette analyse de la coopération entre utilisateurs et informaticiens dans l'entreprise, nous souhaitons souligner les points suivants :

1. L'hypothèse de la théorie des coûts de transaction est que l'organisation interne est un mode de gouvernance favorisant l'adaptation coopérative.

2. Cette hypothèse est contredite dans notre domaine particulier d'application de la théorie par un ensemble de facteurs limitant la coopération entre utilisateurs et informaticiens (style cognitif, caractéristiques des tâches et critères d'évaluation, cadre de référence et système de loyauté, situation dans la structure organisationnelle).

3. Des actions peuvent être entreprises pour réduire l'impact de ces facteurs :

- les nouveaux modes de développement, en particulier le prototypage, permettent une communication plus intuitive et heuristique,

- les trois derniers facteurs sont liés au caractère technique de la fonction ; si l'informaticien interne est perçu uniquement comme un spécialiste technique, il a peu de chances d'établir un dialogue satisfaisant avec les autres membres de l'organisation ; les actions à entreprendre doivent donc viser à développer les autres dimensions de son activité.

4. la communication entre utilisateurs et informaticiens semble cependant s'améliorer dans les dernières années ; tout autant que la participation effective au développement c'est l'implication des responsables qui est la clé d'une valorisation efficace de l'informatique interne.

5. *Si ces conditions sont réunies*, l'hypothèse d'adaptation coopérative interne formulée par Williamson peut s'appliquer.

Nous avons retrouvé avec cette analyse de la coopération le poids des facteurs environnementaux ("l'atmosphère") dans le modèle de Williamson. Nous allons maintenant revenir sur le coeur de la problématique des coûts de transactions, c'est-à-dire les attributs de la transaction.

II - Attributs des transactions : spécificité des actifs et difficultés d'évaluation

2.1 Spécificité des actifs

Quels sont les actifs spécifiques que doit éventuellement engager un fournisseur pour une transaction de développement de logiciel ? La spécificité géographique tend à perdre de l'importance avec le développement des télécommunications et la télémaintenance. Des expériences de sous-traitance du développement à l'étranger ont ainsi pu être menées par des SSII et apparemment couronnées de succès (Ravichandran et Ahmed, 1993). La spécificité physique (matériel) est de moins en moins évidente du fait du développement de la portabilité et de la compatibilité entre systèmes. Pour le développement de logiciel, les actifs spécifiques sont surtout humains et correspondent à la constitution d'un savoir-faire au cours de la transaction.

Le développement de logiciel met en jeu pour les informaticiens plusieurs types de savoir-faire, correspondant à différentes tâches :

- compréhension de l'activité de l'utilisateur et du problème à informatiser,
- traduction en termes techniques du besoin (conception de l'application informatique),
- maîtrise de la technologie de développement (base de données, AGL, système expert...),
- intégration de l'application dans le système informatique actuel,
- gestion de projets.

Le savoir-faire technologique est facilement redéployable pour un fournisseur externe (si une maîtrise particulière d'un outil a été mise en oeuvre, elle sera réutilisée pour d'autres clients). La plupart des SSII sont ainsi spécialisées dans une gamme de langages et d'outils et refusent de développer pour un client une nouvelle connaissance technologique si cela ne correspond pas à leur propre politique de produits et si elles ne pensent pas pouvoir à terme redéployer cette connaissance auprès d'autres clients.

La gestion de projet met surtout en oeuvre des techniques du métier (méthodes de planification, suivi, indicateurs de contrôle) qui sont communes à tous les projets. Ce savoir faire peut donc être assez facilement redéployable.

En revanche, le fournisseur devra investir beaucoup en phase d'analyse du besoin, constituer des liens avec les utilisateurs, mener l'étude spécifique du problème et ne pourra que très partiellement réutiliser cette connaissance auprès d'autres clients. Ceci est particulièrement vrai si l'activité est très spécifique à l'entreprise (marché limité, niches) ou encore celle-ci a un mode de gestion de l'activité qui lui est propre et auquel elle tient. L'investissement spécifique est plus réduit pour le fournisseur s'il correspond à une spécialisation horizontale par type d'application (gestion du personnel, système de reporting ou de consolidation..) ou verticale (spécialisation sur un secteur d'activité particulier) et d'autant plus que l'on se rapproche des progiciels qui sont l'exemple même d'un savoir-faire applicatif non spécifique.

De même, le savoir-faire déployé lors de l'intégration (connaissance du système d'information pour savoir comment intégrer la nouvelle application) n'est pas redéployable en dehors de l'organisation (mais peut l'être sur d'autres projets pour la même organisation).

Au total, la spécificités des actifs engagés nous semble varier en fonction de deux groupes de facteurs : d'une part les attributs de chaque projet et d'autre part l'étape du processus de développement. Nous allons maintenant revenir en détail sur chacun de ces deux éléments.

2.1.1 Une spécificité des actifs engagés variable selon les différents projets

La spécificité des actifs engagés dans un projet de développement est donc pour nous principalement de deux ordres :

- un premier type d'actifs spécifiques peut être mis en jeu si le problème est peu courant ou si l'entreprise a une façon spéciale de l'aborder ; les actifs spécifiques sont alors liés au domaine d'application du projet (une comptabilité est moins spécifique qu'une application de suivi d'une activité complexe et rare),

- un deuxième type d'actifs spécifiques peut être engagé dans toute la partie interfaces du projet, pour laquelle il est indispensable d'avoir une connaissance approfondie du système d'information actuel de l'organisation ; les actifs spécifiques sont liés à l'intégration du projet dans le système d'information de l'organisation.

En revanche, nous considérons que ni le savoir-faire technologique ni le savoir-faire en gestion de projet ne constituent en règle générale des actifs spécifiques, dans la mesure où les connaissances mises en oeuvre peuvent être assez facilement redéployées par le fournisseur à l'occasion d'autres contrats ou auprès d'autres clients.

La théorie des coûts de transaction nous suggère ainsi l'utilisation de structures de gouvernance différenciées **en fonction des caractéristiques du projet** :

- pour les projets à forte spécificité des actifs (spécificité du domaine d'application ou forte intégration au système d'information), le mode d'organisation interne devrait être privilégié,
- dans les autres cas, le critère de spécificité des actifs permet d'envisager le recours à des structures de gouvernance par le marché ou des formes hybrides.

Cependant la spécificité des actifs ne dépend pas uniquement des caractéristiques de chaque projet ; elle est également variable selon les étapes du développement.

2.1.2 Une spécificité des actifs engagés variable selon les étapes du développement

Pour illustrer cette variation des actifs spécifiques engagés nous avons étudié les caractéristiques de trois étapes classiques du développement (analyse du besoin, conception, réalisation), en nous appuyant en particulier sur les travaux de l'AFAI (1993). Ces caractéristiques sont décrites dans le tableau 6.6 ; elles conduisent à formuler des hypothèses de choix d'externalisation différentes en fonction des étapes du développement d'un même projet.

Etapes du processus de développement			
	Analyse du besoin	Conception	Réalisation
Type de compétence attendue de la part des informaticiens	<u>Compétence managériale</u> identification des opportunités stratégiques <u>Compétence relationnelle</u> liens avec utilisateurs <u>Compétence organisationnelle</u> gestion du changement <u>Assistance à la maîtrise d'ouvrage</u> choix du maître d'oeuvre	<u>Assistance à la maîtrise d'ouvrage</u> choix des moyens, spécifications fonctionnelles et définition des exigences techniques <u>Maîtrise d'oeuvre</u> conception technique, organisation générale de la réalisation, contrat de réalisation	<u>Compétence technique</u> maîtrise des outils <u>Maîtrise d'oeuvre</u> coordination de la réalisation, contrôle continu de l'avancement, contrôle du résultat, encadrement des programmeurs
Connaissances nécessaires	Connaissance du métier et de l'organisation interne	Connaissance de la gestion de projets informatiques et des outils, connaissance de l'organisation et du système d'information actuel	Connaissance de la gestion de projets informatiques, expérience approfondie des outils
Spécificité des actifs	<u>Très forte</u> investissement dans la connaissance de l'organisation, établissement de liens à long terme	<u>Moyenne</u> spécialisation par métier redéployable, investissement dans la connaissance du système d'information interne spécifique	<u>Faible</u> connaissance de la gestion de projets et des outils redéployable, prise de connaissance du projet (résultat de l'analyse du besoin et de la conception) non redéployable
Hypothèses de choix d'organisation	Internalisation	Contrats mixtes, solutions hybrides	Externalisation ou formes hybrides

Tableau 6.6. Caractéristiques des différentes étapes du processus de développement et hypothèses de choix d'organisation dans le modèle des coûts de transaction

L'étape d'**analyse du besoin** implique la connaissance de l'organisation interne ainsi que le développement de liens relationnels à long terme avec les dirigeants et les responsables utilisateurs. Ces différents éléments sont difficilement redéployables pour un fournisseur extérieur. Elle se caractérise ainsi par une spécificité des actifs très élevée, ce qui conduit selon les hypothèses de la théorie des coûts de transaction, à privilégier le recours au mode de gouvernance interne.

L'étape de **conception** met en jeu à la fois des actifs :

- facilement redéployables : connaissances techniques, modélisation et gestion de projet,
- redéployables seulement sous certaines conditions : connaissance du métier exercé par l'organisation (réutilisable dans le cadre d'une spécialisation sectorielle),
- spécifiques : connaissance du système d'information actuel de l'organisation.

L'étape de conception correspond à une spécificité des actifs "moyenne", ce qui conduit, selon la théorie des coûts de transaction, à privilégier le recours aux formes hybrides de gouvernance.

L'étape de **réalisation**, une fois la conception délivrée, met en jeu des actifs assez facilement redéployables sur le marché : compétence technique, maîtrise des outils, maîtrise d'oeuvre du projet. L'hypothèse que l'on peut formuler pour cette étape à partir de la théorie des coûts de transaction est donc celle du recours au marché.

A noter que la différenciation entre les étapes du processus sera plus ou moins accentuée selon les projets : un domaine d'application peu spécifique limitera les investissements nécessaires en phase d'analyse du besoin, un projet très intégré au système d'information exigera des études très spécifiques en phase de conception.

2.2 Difficultés d'évaluation du bien

Nous avons vu dans le chapitre précédent que la difficulté d'évaluation du bien est l'un des problèmes majeurs posés par le logiciel. Plusieurs caractéristiques concourent à cette difficulté:

- la dépendance du logiciel à son contexte d'utilisation fait qu'il est très difficile d'évaluer *a priori* et hors situation réelle la qualité du produit et sa valeur ajoutée pour l'utilisateur,
- le succès de l'implantation du logiciel et son adaptation au contexte dépendent d'ailleurs à la fois du fournisseur de la ressource et de l'utilisateur ; en ce sens, c'est une production jointe,
- les délais de réalisation reportent l'évaluation dans le temps,
- les problèmes de communication entre informaticiens et utilisateurs engendrent des risques supplémentaires.

Les nouveaux paradigmes de développement devraient permettre d'assurer une meilleure cohérence entre l'expression du besoin et le logiciel produit et améliorer, avec des techniques comme le prototypage, la visibilité *a priori* sur le résultat. (Whang, 1992). Cependant, nous considérons que la difficulté d'évaluation du bien est pour l'instant encore l'un des handicaps majeurs à une organisation des transactions par le marché ; dans l'incapacité de mettre en place un contrôle par le résultat, beaucoup d'entreprises préféreraient conserver un contrôle continu du comportement du fournisseur, ce qui implique une organisation interne ou pour le moins hybride.

Comme pour la spécificité des actifs, ce critère est variable selon les projets et les étapes du développement :

- un **projet très structuré** permet en règle générale une meilleure visibilité *a priori* sur le résultat final,
- l'évaluation du résultat devient plus facile au fur et à mesure que l'on progresse dans le projet ; en particulier, la **difficulté d'évaluation du bien est moins élevée en phase de réalisation qu'en phase d'analyse du besoin ou même de conception** ; elle peut être considérablement réduite en phase de réalisation si l'on utilise des techniques de prototypage dans les étapes précédentes.

Conclusion

En conclusion sur cette partie consacrée à la théorie des coûts de transaction, nous pouvons souligner les éléments suivants :

L'analyse des caractéristiques juridiques du logiciel faite dans le chapitre précédent avait mis en évidence un ensemble d'éléments qui éloignent le logiciel de la définition classique des biens échangés sur des marchés parfaits : difficulté d'évaluation du bien, dépendance au contexte d'utilisation, caractère de production jointe requérant une coopération des parties, asymétrie d'information potentielle entre un utilisateur plus ou moins averti et un spécialiste d'une technologie complexe et toujours en évolution, délais longs, risque potentiel élevé. Ces caractéristiques sont généralement très accentuées dans le cas de logiciels spécifiques mais sont sujettes à variation en fonction de la nature des projets.

La théorie des coûts de transaction nous a permis de distinguer trois structures de gouvernance (formes contractuelles) majeures que l'on peut observer en pratique dans le cas de l'activité de développement d'applications : la hiérarchie (développement interne), le marché (développement au forfait), les formes hybrides (régie et combinaison de différentes solutions contractuelles pour le même projet).

Le choix de la structure de gouvernance adaptée dépend principalement, selon Williamson, des attributs de la transaction (spécificité des actifs et difficulté d'évaluation du bien). Ces éléments varient :

- selon les caractéristiques de chaque **projet**

Nous retenons trois facteurs associés au projet. Deux sont liés à la spécificité des actifs : le domaine d'application et le degré d'intégration du projet dans le système informatique actuel. Le troisième facteur est la difficulté d'évaluation du bien, qui correspond à la difficulté d'estimation *a priori* par l'utilisateur de la qualité du logiciel ; celle-ci dépend de plusieurs critères parmi lesquels nous avons retenu le degré de structure du projet.

- en fonction des **étapes du processus de développement**

Nous avons étudié trois étapes classiques du processus : l'analyse des besoins, la conception et la réalisation. Celles-ci se différencient tant sur le critère de spécificité des actifs que sur celui de facilité d'évaluation du bien, ce qui conduit à formuler des hypothèses différenciées de choix de mode d'organisation pour chacune d'entre elles.

Une interrogation demeure cependant sur l'applicabilité de la théorie des coûts de transaction à notre problème : celle-ci est liée à la capacité de coopération entre informaticiens internes et utilisateurs.

Le tableau 6.7. reprend la synthèse de ces différents éléments.

Principaux éléments liés aux coûts de transaction	Application à l'activité de développement d'applications	Difficultés et limites
<p>La théorie des coûts de transaction associe de façon sélective "des transactions, qui diffèrent dans leurs attributs et des structures de gouvernance, qui diffèrent dans leurs coûts et leurs capacités, sur le critère principal des coûts de transaction".</p> <p>Les coûts de transaction sont des coûts d'organisation et de contrôle de la relation contractuelle.</p> <p>Le principal attribut des transactions est la spécificité des actifs, la difficulté d'évaluation du bien doit également être prise en compte.</p> <p>Les structures de gouvernance se différencient par leurs capacités respectives d'adaptation, le régime d'administration des contrats et les systèmes de pilotage et de contrôle.</p>	<p>Les trois structures de la gouvernance de Williamson sont utilisées pour le développement d'applications spécifiques :</p> <ul style="list-style-type: none"> - hiérarchie (développement interne), - marché (forfait), - formes hybrides (régie et solutions mixtes). <p>Les projets et les étapes du processus se différencient par une spécificité plus ou moins grande des actifs engagés et une évaluation du résultat <i>a priori</i> plus ou moins aisée.</p>	<p>Certains facteurs peuvent remettre en cause le portrait-type des capacités de l'organisation interne (hiérarchie) défini par Williamson.</p> <p>En particulier, la capacité d'adaptation interne ne sera effective que si les relations utilisateurs informaticiens sont satisfaisantes ; des études récentes semblent montrer une évolution en ce sens.</p> <p>La qualité de ces relations dépend des capacités des informaticiens internes à développer les aspects non techniques de leur tâche (orientation "managériale").</p>

Conditions de mise en oeuvre de l'externalisation	Variables à prendre en compte dans la décision d'externalisation
<p>Pour bénéficier des avantages du marché, il faut que le nombre de fournisseurs soit suffisant (concurrence réelle).</p>	<p>Concernant le projet :</p> <ul style="list-style-type: none"> • spécificité des actifs <ul style="list-style-type: none"> - domaine concerné - niveau d'intégration au système d'information actuel • difficulté d'évaluation <i>a priori</i> <p>Concernant les étapes :</p> <ul style="list-style-type: none"> • spécificité des actifs et difficulté d'évaluation du logiciel plus fortes au début du projet.

Tableau 6.7 Synthèse de l'approche des coûts de transaction

Chapitre 7 :
Synthèse des approches théoriques -
modèle et hypothèses de la recherche empirique

La revue de la littérature faite dans les chapitres précédents nous a permis de donner différents éclairages du problème d'arbitrage entre ressource interne et externe, correspondant à différentes approches théoriques de l'externalisation. Chacune de ces approches théoriques a fourni un éclairage particulier et a permis de mettre en évidence l'importance de variables dans le choix.

L'objectif de ce chapitre est, au travers de la synthèse de ces différentes approches, de construire un modèle global et d'élaborer un ensemble d'hypothèses à tester lors de la recherche empirique.

Ce chapitre est organisé en deux sections. La première présente une synthèse des principaux apports de l'analyse théorique, en rappelant notamment les hypothèses et facteurs de choix associés à chacune des quatre approches. La seconde décrit et explique la construction du modèle et des hypothèses de la recherche empirique.

Section 1 - Les principaux apports de l'analyse théorique

Nous reprenons ici rapidement chaque approche, pour rappeler les principales conclusions dégagées dans les quatre chapitres précédents.

I. Les enseignements de l'approche de la spécialisation industrielle

Cette approche conduit à choisir l'externalisation à chaque fois que la spécialisation industrielle présente des avantages en termes de coûts de production. Elle est centrée sur l'optimisation opérationnelle du processus, la recherche d'économies d'échelle et d'étendue et l'adéquation charge/capacité. La technologie y joue un rôle essentiel, car elle détermine (avec le volume de production) les économies d'échelle ou d'étendue qui permettent de réduire les coûts de production. La rupture technologique actuelle pourrait ainsi justifier une évolution des choix d'externalisation. Gérer cette rupture en interne pose des problèmes d'investissement aux organisations : sélection des nouvelles technologies pertinentes pour l'entreprise et coût d'acquisition des compétences correspondantes. Sur un plan opérationnel, la principale difficulté est l'inexpérience des équipes internes, qui est source d'incertitude sur la capacité de l'entreprise à maîtriser, seule, des projets à forte composante technologique.

Appliquée à notre domaine, l'approche industrielle permet de formuler les hypothèses suivantes :

- A un niveau global (gestion de l'activité), les choix d'externalisation dépendent d'une part des perspectives sur la pérennité de la demande interne en logiciels spécifiques et d'autre part de l'état de la technologie.

- Concernant la gestion des projets, les choix d'externalisation dépendent de critères opérationnels : capacité disponible en interne, délais de production, connaissance de la technologie particulière utilisée pour le projet et intégration du projet au système d'information actuel (rappelons que ce dernier critère est à notre avis défavorable au fournisseur).

Rappelons enfin que l'approche industrielle concerne surtout les étapes techniques du processus (programmation et, à un moindre degré, conception).

II. Les enseignements de l'approche stratégique (théorie de la ressource)

L'approche stratégique insiste sur la cohérence des activités exercées par l'entreprise et la stratégie globale. L'affectation des ressources internes doit être en priorité dirigée vers les activités présentant un véritable apport de valeur ajoutée. Les autres activités, qui ne présentent pas un enjeu vital et n'offrent pas de perspective de gain d'avantage concurrentiel, peuvent être externalisées et doivent même l'être si l'on raisonne en coût d'opportunité et capacité de développement des ressources de base de l'organisation.

Les hypothèses et facteurs de choix associés à cette approche sont les suivants :

- Les choix d'externalisation dépendent de l'intérêt stratégique de l'activité de développement d'applications, tel qu'il est perçu dans l'organisation.
- L'intérêt stratégique dépend à un niveau global (gestion de l'activité) du secteur d'activité et du type de stratégie adoptée par l'entreprise,
- L'intérêt stratégique est variable selon les projets au sein du portefeuille d'applications,
- L'intérêt stratégique est variable selon les étapes du processus de développement. A cet égard, nous formons l'hypothèse que l'étape d'analyse du besoin présente un intérêt stratégique fort pour l'organisation, à la fois parce qu'elle s'avère une étape critique du processus et parce qu'elle est le moment du projet où s'exprime la compétence architecturale.

III. Les enseignements de la théorie de l'agence

La théorie de l'agence permet de mettre en évidence les difficultés et les risques de la relation contractuelle associée à la situation de délégation d'une tâche. Elle distingue deux grands modes de pilotage de la relation contractuelle, appuyés l'un sur l'évaluation du comportement, l'autre sur l'évaluation du résultat.

Les hypothèses et facteurs de choix associés à ce modèle sont les suivants :

- Le choix entre les deux modes de pilotage dépend principalement de la possibilité de mettre en oeuvre au moindre coût l'un ou l'autre mode de pilotage (possibilité de contrôle du résultat et possibilité de contrôle du comportement).

- La possibilité de contrôle du comportement est meilleure si la relation entre les deux parties est établie sur une durée longue ; elle nécessite la mise en oeuvre de systèmes de contrôle coûteux.

- La possibilité de contrôle du résultat dépend en partie des caractéristiques du projet : projet très structuré ou non *a priori* (cahier des charges complet, prototypage), processus de production bien maîtrisé ou non sur le plan technique.

- L'incertitude sur le résultat est très forte en début de projet (étape d'analyse du besoin) ; elle diminue en cours de processus, ce qui laisse envisager la possibilité de faire varier le mode de pilotage en fonction des étapes du développement.

IV. Les enseignements de la théorie des coûts de transaction

L'hypothèse centrale de cette approche est l'adéquation entre les attributs des transactions et le mode d'organisation ("structure de gouvernance") retenu. L'objectif est de minimiser la somme des coûts de production et de transaction, sachant que ceux-ci sont déterminés à la fois par les attributs des transactions et par les aptitudes des différents modes d'organisation, qui leur permettent d'être plus ou moins efficaces pour gérer tel ou tel type de transaction. Dans le domaine informatique, on trouve couramment les trois modes d'organisation : le forfait, qui correspond au marché, l'informatique interne, qui est la forme hiérarchique, la régie et les combinaisons de ressources, qui sont des formes hybrides.

Les hypothèses et facteurs de choix associés à ce modèle sont les suivants :

- Sur un plan global, l'efficacité du mode interne sur le critère des coûts de transaction dépend de la qualité de l'adaptation coopérative entre utilisateurs et informaticiens ; celle-ci est

principalement liée à la capacité de ces derniers à développer l'orientation managériale de leurs tâches.

- Les choix d'externalisation concernant les projets sont faits en fonction de la spécificité des actifs engagés par les différents projets (spécificité du domaine d'application et le degré d'intégration du projet au système d'information existant). Le critère de facilité d'évaluation du bien développé par Barzel nous semble également pertinent dans le cas du logiciel ; il varie en fonction des caractéristiques des projets (projet très structuré ou non *a priori*).

- Les choix d'externalisation dépendent également des caractéristiques de chaque étape du processus ; la spécificité des actifs et la difficulté d'évaluation du logiciel sont plus fortes au démarrage du projet (étape d'analyse du besoin).

	Approche stratégique	Approche industrielle	Théorie de l'agence	Théorie des coûts de transaction
Enseignement de chaque approche	Les choix d'externalisation sont liés à l'importance stratégique : l'entreprise gère en interne les activités stratégiques et externalise les autres activités	Les choix d'externalisation sont opérés sur le critère des coûts de production. Le volume d'activité et la technologie sont les principaux déterminants de ce coût.	Le choix du mode de pilotage correspond au mode de contrôle le plus efficient : - contrôle du comportement - contrôle du résultat.	Le choix du mode de pilotage dépend : - de la spécificité des actifs engagés - de la difficulté d'évaluation du résultat <i>a priori</i> - des capacités des différents modes d'organisation

Tableau 7.1. Synthèse des enseignements des différentes approches

Section 2 - Modèle et hypothèses de la recherche empirique

A l'issue de l'analyse théorique nous avons pu identifier un ensemble de facteurs ; la première constatation a été l'existence de différents niveaux d'analyse des choix d'externalisation. Nous avons retenu trois groupes de facteurs, intervenant à trois niveaux différents (tableau 7.2). Certains facteurs, comme l'importance stratégique, sont déclinés aux trois niveaux ; d'autres n'interviennent qu'à un seul plan.

1. Les premiers facteurs concernent l'activité de développement d'applications, à un niveau global, correspondant au niveau "stratégique" dans le cadre d'analyse de Rands (cf. chapitre 4). Ils influent sur la **politique de choix de ressources à long terme**.

- Le secteur d'activité et le type de stratégie adoptée par l'entreprise conditionnent l'importance stratégique accordée à l'activité ; moins celle-ci est élevée, plus l'entreprise aura tendance à externaliser l'activité.

- La pérennité de la demande interne en logiciels spécifiques et l'évolution de la technologie conditionnent la capacité de l'entreprise à obtenir en interne des coûts de production comparables à ceux des fournisseurs extérieurs ; quand la demande baisse, le maintien de l'activité en interne devient économiquement difficile à justifier ; la rupture technologique actuelle pourrait quant à elle conduire à une "logique de spécialisation" favorable à l'externalisation.

- La durée de la relation correspond à la façon dont l'entreprise envisage le contact avec les fournisseurs (interne ou extérieurs) en logiciels spécifiques ; souhaite-t-elle avoir un "partenaire privilégié à long terme" ou envisage-t-elle au contraire une série de contrats séparés avec différents fournisseurs ? L'hypothèse est que le mode de pilotage approprié dans le premier cas correspond à une évaluation du comportement (interne/régie) et dans le second cas à une évaluation du résultat (forfait).

- La qualité de la coopération entre les informaticiens internes et les utilisateurs est le dernier critère de ce groupe. Elle dépend principalement du développement d'une orientation

managériale chez les informaticiens. L'hypothèse est que l'entreprise aura davantage tendance à externaliser l'activité de développement d'applications si la coopération interne est faible.

2. Le second groupe de facteurs est lié aux caractéristiques des **projets**. Ce sont en majorité des facteurs opérationnels, correspondant au niveau **tactique** des décisions, selon le modèle de Rands.

- La nouveauté technologique du projet, les problèmes d'ajustement charge/capacité, les délais et l'intégration du projet au système d'information sont les critères opérationnels qui conditionnent les choix d'externalisation. Une forte nouveauté technologique, une capacité interne insuffisante, des délais serrés et une faible intégration du projet au système d'information sont à notre avis favorables à l'externalisation.

- Le degré de structure du projet conditionne la capacité d'évaluer le résultat a priori. L'hypothèse est que l'entreprise a davantage tendance à externaliser des projets très structurés.

- La spécificité du domaine et l'intégration du projet au système d'information sont deux facteurs de spécificité des actifs engagés pour le projet. L'hypothèse est que l'entreprise a intérêt à externaliser des projets correspondant à une faible spécificité des actifs engagés.

- Enfin, nous formons l'hypothèse que l'importance stratégique du projet peut également jouer à ce niveau, comme élément d'arbitrage entre différents projets ; l'organisation externalise les projets jugés peu stratégiques.

3. Le troisième niveau de décision correspond aux choix d'externalisation pour les différentes **étapes** du processus de développement. Nous formulons l'hypothèse d'une différenciation des choix selon les étapes, en fonction de leurs caractéristiques stratégiques et industrielles et du mode de contrôle le mieux adapté à chacune d'entre elles :

- Internalisation privilégiée pour l'étape d'analyse du besoin.
- Choix de l'externalisation (forfait ou régie) pour l'étape de conception.
- Externalisation (au forfait) de l'étape de réalisation.

Facteurs déterminant les choix d'externalisation				
3 niveaux	stratégiques	industriels	agence (*)	coûts transaction
L'activité de développement (niveau global)	<ul style="list-style-type: none"> • Impact stratégique de l'informatique dans le secteur d'activité (-) • Stratégie de l'entreprise 	<ul style="list-style-type: none"> • Evolution de la technologie (+) • Pérennité de la demande (-) 	<ul style="list-style-type: none"> • Durée de la relation (+) ⇒ contrôle du comportement 	<ul style="list-style-type: none"> • Qualité de la coopération informaticiens / utilisateurs (-)
Les projets (niveau tactique)	<ul style="list-style-type: none"> • Importance stratégique du projet (-) 	<ul style="list-style-type: none"> • Nouveauté technologique (+) • Ajustement charge / capacité (+) • Délais (+) • Intégration au système d'information (-) 	<ul style="list-style-type: none"> • Structure du projet (+) ⇒ contrôle du résultat 	<ul style="list-style-type: none"> • Spécificité du domaine (-) • Intégration au système d'information (-) • Structure du projet (+)
Les étapes du processus	<ul style="list-style-type: none"> • Importance stratégique de chaque étape... 	<ul style="list-style-type: none"> • Avantages liés à la spécialisation du fournisseur... 	<ul style="list-style-type: none"> • Mode de contrôle le plus efficient (comportement / résultat)... 	<ul style="list-style-type: none"> • Spécificité des actifs... • Difficulté d'évaluation du résultat <i>a priori</i>...
----- Analyse du besoin	...forte : ⇒ internalisation	...faibles : ⇒ internalisation	...comportement : ⇒ interne ou régie	...fortes : ⇒ internalisation
----- Conception	...faible : ⇒ externalisation	...moyens : ⇒ externalisation		...moyennes : ⇒ forme hybride
----- Réalisation	...faible : ⇒ externalisation	...forts : ⇒ externalisation	...résultat : ⇒ forfait	...faibles : ⇒ forfait

Tableau 7.2. Modèle global et hypothèses de la recherche empirique

Un (-) signifie que nous faisons l'hypothèse d'une corrélation négative entre ce facteur et l'externalisation.
exemple : moins l'entreprise croit à la pérennité de la demande interne, plus elle aura tendance à externaliser.
 Un (+) signifie que nous faisons l'hypothèse d'une corrélation positive entre ce facteur et l'externalisation.
exemple : plus le projet est structuré, plus l'entreprise aura tendance à l'externaliser.

(*) La théorie de l'agence ne permet pas d'inférer directement les choix d'externalisation mais d'émettre des hypothèses sur le mode de contrôle le plus adapté (comportement ou résultat) ; nous avons par ailleurs rappelé ici l'association classique entre d'une part le mode interne et la régie et un contrôle plutôt axé sur l'observation du comportement, et d'autre part le forfait et un contrôle plutôt axé sur l'observation du résultat.

En conclusion sur le modèle et les hypothèses de la recherche empirique, nous souhaitons souligner les points suivants :

- Le modèle construit à partir de l'analyse théorique apparaît relativement complexe et extrêmement contingent ; il met en évidence un nombre élevé de facteurs, intervenant à différents niveaux. Parvenus à ce stade de nos réflexions, nous nous sommes posé la question suivante : avons nous retenu *trop* de facteurs ? Etait-il possible d'aboutir à un modèle plus simple, en éliminant tel ou tel aspect secondaire ? Nous avons choisi de ne pas simplifier le modèle obtenu, ce pour deux raisons. Tout d'abord, chacun des facteurs est le fruit d'une analyse théorique qui a conduit à l'identifier ; choisir entre différents facteurs pour les hiérarchiser et n'en retenir que quelques uns comportait une part d'arbitraire difficilement justifiable et que nous ne souhaitons pas introduire dans ce travail. La seconde raison est liée au caractère exploratoire de la recherche : en l'absence de résultats empiriques disponibles *a priori*, nous ne pouvons pas écarter tel ou tel facteur dont nous ne savons pas s'il joue en pratique sur les choix d'externalisation.

- Un élément essentiel de notre modèle est la **différenciation des choix en fonction des étapes du processus**. Cette différenciation, qui découle logiquement de l'analyse théorique, est cependant une "nouveau" dans la littérature du domaine des systèmes d'information. Comme nous l'avons montré dans les premiers chapitres de ce travail, les auteurs du domaine envisagent les choix d'externalisation soit de façon globale, en parlant de l'activité de développement d'applications et sans en différencier les différentes composantes que sont les étapes du processus, soit en faisant référence aux différents projets. Nous pensons au contraire que la réflexion sur les caractéristiques des différentes étapes du processus, sur les compétences engagées et sur les modes de contrôle appropriés est une piste essentielle pour comprendre et améliorer le mode de gestion de l'activité.

- Le choix d'un exposé en chapitres distincts des quatre approches étudiées a conduit à privilégier les facteurs de différenciation plutôt que les points communs entre les différentes approches. Le modèle permet cependant de mettre en évidence des liens et des éléments convergents. Par exemple, il est clair que l'évolution de la technologie joue à la fois sur la formation des coûts de production *et* sur les capacités de contrôle de l'activité : plus la production est industrialisée et meilleure est la visibilité *a priori* sur le résultat ; l'utilisation du prototypage en phase d'analyse du besoin améliore à la fois les conditions de production des étapes suivantes (définition plus précise, tests menés en début de projet) et le degré de confiance que l'utilisateur peut avoir dans la conformité du résultat à ses attentes. Autre exemple, le degré d'intégration d'un nouveau projet au système d'information existant joue sur l'avantage comparatif des équipes internes et externes sur le critère des coûts de production ; mais il joue également sur la spécificité des actifs engagés par le fournisseur et s'avère ainsi un facteur susceptible d'influer sur les coûts de transaction. Enfin, la convergence la plus frappante entre les différentes approches concerne les hypothèses de choix d'externalisation associées aux différentes étapes du processus. Celles-ci conduisent, pour des raisons distinctes, à des choix cohérents, ce qui traduit le fait qu'elles ne sont que différentes facettes du même phénomène.

Troisième partie : Etude empirique

Chapitre 8 :
Résultats de la recherche quantitative

Introduction

Nous allons détailler dans ce chapitre les résultats de la recherche quantitative pour laquelle nous avons collecté, sur la base d'un questionnaire, des données concernant 60 projets applicatifs initiés ou mis en oeuvre dans les années 1992 et 1993. La présentation de l'échantillon et des variables étudiées est faite dans la première section du chapitre.

Notre objectif dans ce chapitre est double :

- Tout d'abord, nous cherchons à **décrire** les pratiques existantes. Nous avons en effet pu constater le manque de recherches descriptives sur ce thème, ce qui fait que l'on dispose de très peu d'informations, en dehors de chiffres globaux indiquant le chiffre d'affaires par nature de prestation (cf. les études citées dans le chapitre 1). Nous cherchons donc à approfondir la connaissance du comportement des entreprises quant à leur choix de ressources et à décrire ce comportement de façon structurée. De ce fait, nous consacrons la seconde section à une analyse descriptive de nos variables "ressources utilisées", en utilisant différentes méthodes statistiques (étude des corrélations, analyse en composantes principales, typologie). Cette analyse fait une large part à l'étude de la variation des choix d'externalisation en fonction de **l'étape du processus**.

- Nous cherchons ensuite à déterminer dans quelle mesure les comportements observés corroborent ou non les hypothèses de notre modèle, en nous centrant ici sur les facteurs liés au **projet**. Nous avons donc demandé à nos interlocuteurs de préciser pour chaque projet deux groupes de variables : un premier groupe est formé de variables descriptives du projet (taille, urgence, etc.), un second de variables correspondant à un jugement sur les raisons qui ont conduit à privilégier un certain choix de ressources pour le projet considéré. Nous étudions dans la troisième section de ce chapitre le lien entre ces variables avec la typologie de projets élaborée à partir des choix de ressources précédemment analysées.

L'étude quantitative n'aborde pas les facteurs environnementaux jouant au niveau global de l'activité ; ceux-ci sont analysés dans le chapitre 9.

Section 1 - Définition de l'échantillon ; présentation des variables et des hypothèses détaillées

Conformément aux hypothèses de notre modèle, notre collecte de données quantitatives a été effectuée sur la base d'un questionnaire par projet de développement d'application, détaillant les différentes étapes du processus et dont un exemplaire est fourni en annexe. Compte tenu du caractère exploratoire de la recherche, nous ne pouvions pas prétendre dans cette analyse quantitative à valider les hypothèses de notre modèle ; les objectifs principaux sont la description des pratiques et une première tentative de corroboration (forme faible de validation) ou d'infirmer des hypothèses.

I. Critères de choix des entreprises et des projets

Nous avons collecté l'information sur les choix de ressources détaillés par étapes pour 60 projets réalisés ou mis en oeuvre dans les deux dernières années. Il était en effet important pour nous de collecter des informations récentes, sachant que les comportements ont pu évoluer dans les dernières années. Seuls les projets achevés en cours ont été pris en compte, excluant ainsi la collecte d'informations prospectives sur le portefeuille d'applications restant à mettre en oeuvre, que nous avons jugé trop peu fiables pour être intégrées à l'étude.

Il eût été envisageable de collecter l'information dans une seule grande entreprise ; c'est la méthode qui a été adoptée par exemple par Walker et Poppo (1991) qui ont étudié les choix de ressources pour 67 des 250 composants de la division assemblage d'une grande entreprise de fabrication américaine. Cependant dans notre cas il paraissait difficile de trouver dans la même entreprise un nombre suffisant de projets de développement spécifique respectant nos critères de taille (élimination des petits projets inférieurs à un mois de développement) et d'ancienneté (projets des deux dernières années). Cela nous aurait amené à nous intéresser au cas des très grandes entreprises pour lesquelles l'informatique se situe au coeur du métier (banque, assurance), ce qui n'était pas notre objectif.

En effet, nous avons voulu examiner la situation, plus fréquente, des entreprises pour lesquelles l'informatique et le développement d'application sont des ressources nécessaires, certes, mais pas *a priori* entièrement stratégiques. Nous voulions également examiner les choix de ressources sur un nombre assez important de projets ; ceci nous a conduit à mener notre étude auprès de plusieurs entreprises. Les données ont été collectées auprès de douze entreprises qui nous ont demandé de respecter une certaine confidentialité et que nous ne désignerons donc pas par leur vrai nom. Nous avons pour la même raison indiqué le secteur d'activité général, et non précis, de l'organisation.

Nous avons choisi les entreprises en privilégiant trois critères :

1. Nous avons exclu de la recherche les entreprises appartenant à des secteurs où les systèmes d'information sont au coeur du métier, parce qu'ils jouent un rôle majeur d'une part sur le processus de création de la valeur ajoutée et d'autre part sur le contenu du produit. En particulier, le secteur de la banque et de l'assurance a été exclu.

Cette option répond à plusieurs préoccupations. Tout d'abord, compte tenu du caractère exploratoire de notre recherche et de la multiplicité des facteurs de contingence (stratégiques, industriels, liés au contrôle de l'activité) que l'analyse théorique permettait de dégager *a priori*, il nous a semblé irréaliste de vouloir étudier dans cette première recherche à la fois le cas des secteurs d'activité où l'informatique occupe une position clef et celui d'autres secteurs où elle est moins stratégique. Il fallait donc faire un choix.

Pourquoi le choix de secteurs d'activité où l'informatique est "moins stratégique" ? On pourrait en effet nous opposer qu'il eût été plus intéressant de faire le choix inverse, compte tenu des enjeux plus importants pour les entreprises et de la taille plus grande des services de développement d'applications. C'est en fait précisément l'importance des enjeux qui nous a conduit à écarter ces secteurs d'activité. Nous avons en effet craint qu'elle soit un facteur d'inertie, les entreprises préférant adopter au moins dans un premier temps une attitude prudente d'attente et d'observation avant d'introduire des changements dans la gestion de l'activité. Le choix de secteurs où l'informatique est moins stratégique nous a donc semblé plus propice à l'observation de nouveaux comportements d'externalisation.

Ce choix a des répercussions sur deux autres critères : c'est en effet dans les secteurs où l'informatique est stratégique que l'on trouve les directions informatiques à la fois les plus importantes en taille et les plus puissantes. Nous considérons de ce fait que nos conclusions ne sont pas applicables à ces secteurs, même avec des précautions d'usage ou des restrictions et qu'une autre étude serait nécessaire, qui comporterait sans doute des hypothèses en partie différentes pour tenir compte du poids stratégique et du pouvoir élevé des directions informatiques.

2. Un deuxième critère de choix a été la taille à la fois des entreprises ou des divisions utilisatrices et des services études internes¹. Le choix du niveau (entreprise, division) dépendait des pratiques de centralisation ou de décentralisation des développements dans les directions opérationnelles. Nous avons évité les très grandes tailles (pas d'effectifs utilisateurs supérieurs à 20.000 personnes, pas d'effectifs de services études supérieurs à 100 personnes) comme les petites (pas d'effectifs utilisateurs inférieurs à 800 personnes ; pas de contrainte de taille minimum en revanche sur le service études, un effectif très réduit pouvant être le résultat de pratiques d'externalisation).

3. Un troisième critère a répondu à un objectif pratique d'accessibilité des données. Nous avons ainsi engagé au départ des contacts avec une vingtaine d'entreprises, dont certaines ont été éliminées après analyse sur l'un ou l'autre des deux critères précédents. Nous souhaitons à chaque fois mener un ou plusieurs entretiens qualitatifs approfondis et obtenir un jeu de questionnaires projets permettant de décrire au minimum un exemple des différentes pratiques de choix de ressources. La lourdeur de la procédure a provoqué quelques abandons et nous a conduit au final à retenir 12 entreprises, qui ont mené les 60 projets pour lesquels nous avons obtenu une réponse complète au questionnaire.

Pour toutes entreprises étudiées, l'utilisation de ressources extérieures fait partie de la gestion courante de l'activité de développement d'applications : elles utilisent à la fois des ressources internes et externes. Elles font en général appel - à des degrés divers cependant - à

¹. Ce critère est lié au premier. les très grandes entreprises intervenant dans des secteurs où l'informatique est stratégique ayant les plus gros services internes.

l'ensemble des trois modes contractuels (interne, régie, forfait). Nous avons demandé à chaque fois à nos interlocuteurs de sélectionner les **projets** sur les critères suivants :

- choisir les projets récents et importants pour l'entreprise,
- donner un éventail des choix de ressources effectivement pratiqués par l'organisation.

Ainsi, nous avons privilégié dans la sélection des projets la diversité des choix plutôt que la représentativité. Cela conduit à sur-représenter dans l'échantillon des pratiques peu courantes. C'est à notre avis en particulier le cas du forfait, que la plupart des entreprises interrogées pratiquaient de façon marginale sur quelques projets mais qui, lorsqu'il était utilisé, a systématiquement été inclus dans l'échantillon.

Il nous faut rappeler ici que nous ne prétendons pas à une représentativité de notre échantillon sur les entreprises françaises, ni même sur tel ou tel sous-groupe d'entre elles. Nous n'avions ni les moyens ni le temps nécessaires pour atteindre cet objectif, dont nous pensons de plus qu'il aurait été prématuré compte tenu des connaissances disponibles au départ sur le sujet.

II. Les interlocuteurs rencontrés

Nous voulions rencontrer dans chaque organisation la ou les personnes responsables des choix d'externalisation, appartenant soit au groupe des utilisateurs, soit à l'informatique. En pratique, nous avons cherché à rencontrer les deux types de profils mais dans de nombreux cas il s'est avéré que ce sont les informaticiens qui sont les gestionnaires des ressources, y compris extérieures.

Ainsi, les utilisateurs qui nous ont accordé un entretien général sur les choix d'externalisation se sont dans la quasi-totalité des cas déclarés incompetents pour répondre aux questionnaires détaillés par projet, jugeant que le responsable informatique était la personne à la fois la plus concernée par ces questions et la mieux placée pour y répondre. Ce phénomène est très courant dans les recherches empiriques du domaine des systèmes d'information, où la coopération des utilisateurs est difficile à obtenir, marquant ainsi sans doute un certain retrait de ceux-ci par rapport aux problématiques touchant aux systèmes d'information.

Le tableau 8.1 décrit le profil des personnes rencontrées.

Entreprise	Domaine d'activité ²	Fonction des interlocuteurs
(A)	BTP	- responsable des développements (1) - responsable des projets du domaine GRH (2) - responsable des projets du domaine technique (3) - responsable des projets du domaine commercial (4)
(B)	Chimie	- responsable des études
(C)	Équipementier automobile	- représentant des utilisateurs (1) - responsable des projets du domaine GPAO (2) - responsable des projets du domaine gestion (3)
(D)	Agriculture	- directeur général, en charge de l'informatique
(E)	Industrie d'équipement	- contrôleur de gestion (siège) (1) - responsable études d'une division (2)
(F)	Industrie des métaux	- responsable informatique
(G)	Chimie	- responsable informatique
(H)	Matériaux de construction	- responsable informatique
(I)	Matériaux de construction	- directeur financier (1) - responsable informatique (2)
(J)	Services culturels	- secrétaire général (1) - responsable informatique (2)
(K)	Pétrole	- ancien responsable des applications, ayant repris des fonctions en dehors de l'informatique depuis
(L)	Santé	- responsable informatique

Tableau 8.1. Profil des personnes rencontrées

Nous avons mené avec chacune de ces personnes un entretien qualitatif (les résultats de l'étude qualitative sont présentés dans le chapitre 9 ; les citations sont identifiées sur la base de la lettre associée à chaque cas, complétée le cas échéant d'un numéro d'entretien).

III. Le choix des variables

S'agissant d'un domaine de recherche nouveau, nous avons dû construire nous même les variables opérationnalisant les concepts étudiés. Pour tenter de limiter les risques, nous avons pré-testé notre questionnaire auprès de professionnels du développement d'applications et

². Les entreprises nous ayant en règle générale demandé de respecter une certaine confidentialité, nous avons dans certains cas choisi d'indiquer le secteur de façon relativement générale

administré nous mêmes les questionnaires au cours d'entretiens, ce qui nous a permis de préciser oralement le sens des questions quand cela s'avérait nécessaire.

3.1 Les variables descriptives des choix d'externalisation

Notre objectif était de décrire les choix d'externalisation pour chaque projet.

La première difficulté était d'individualiser les différents projets de l'organisation. En pratique, cela n'a pas posé de problèmes à nos interlocuteurs pour lesquels le découpage des travaux de développement en différents "projets" est usuel. Nous avons donc adopté une définition très empirique du projet, en suivant le découpage opérationnel pratiqué dans les différentes entreprises mais en limitant l'étude à trois étapes du processus (analyse du besoin, conception, réalisation). Les premiers tests du questionnaire ont confirmé la nécessité de dissocier les étapes au sein du projet, les répondants n'associant pas tous le même contenu au terme "développement d'application". En particulier pour certains le "développement" se référait principalement à la phase stricte de programmation alors que pour d'autres elle englobait l'ensemble des étapes du processus³. Pour éviter les confusions, nous avons donc choisi de faire préciser à nos interlocuteurs leurs choix pour chaque étape. Nous voulions également tenter de vérifier notre hypothèse selon laquelle les choix sont différenciés selon les étapes du projet.

L'identification des étapes peut varier selon que l'on se réfère au cycle en cascade ou aux nouveaux modes de développement. Nous avons retenu pour tester notre hypothèse trois étapes⁴ facilement identifiables et présentant des caractéristiques bien différenciées :

- l'analyse du besoin, qui consiste principalement à interagir avec les utilisateurs afin de déterminer ce que doit faire l'application,

³. La confusion était à notre avis accentuée par le fait que nous parlions d'externalisation, phénomène qui concerne en pratique davantage l'étape de programmation.

⁴. Nous n'avons retenu que ces trois premières étapes car elles nous fournissent un éventail de caractéristiques suffisamment diversifié pour tester notre hypothèse de spécialisation par activité. Il serait bien entendu nécessaire dans une démarche exhaustive de prendre en compte les phases de tests, de mise en oeuvre et de maintenance.

- la conception, qui consiste à traduire ce besoin en termes techniques et à intégrer les contraintes du système d'information existant,
- la réalisation, qui est ici la phase de programmation.

Pour qualifier l'externalisation, nous nous sommes référés aux deux contrats classiques dans le domaine du développement (forfait et régie), dont nous avons vu qu'ils correspondent à la fois aux deux modes de pilotage identifiés par la théorie de l'agence (résultat et comportement respectivement) et à deux structures de gouvernance de la théorie des coûts de transaction (marché et forme hybride respectivement). Pour identifier les solutions mixtes, nous avons demandé aux répondants d'indiquer le pourcentage d'appel à la ressource pour chaque étape (tableau 8.2).

Type de ressource utilisée	Etapas du projet		
	Analyse du besoin	Conception	Réalisation
interne	ABinter(%)	Concint(%)	Réainter(%)
régie	ABregie(%)	Concregi(%)	Réaregie(%)
forfait	ABforfai(%)	Concforf(%)	Réaforfa(%)

Tableau 8.2. Variables descriptives des choix d'externalisation pour chaque projet

A noter qu'au sein d'une même étape, les variables descriptives des choix d'externalisation sont liées par construction, puisque l'on a demandé aux personnes interrogées d'indiquer le pourcentage de l'étape réalisé avec chaque moyen, pour un total de 100% (par exemple, $ABinter + ABregie + ABforfait = 100\%$).

Ce mode de description et de collecte des choix d'externalisation a présenté les avantages et inconvénients suivants :

- l'avantage principal a été la **compréhension** immédiate par les répondants de l'outil de mesure, exprimé avec des référents qui leur sont familiers (projet, étapes du processus, modes contractuels généralement pratiqués) ; autre avantage, cette forme les a conduit exprimer leurs

réponses de façon détaillée, sur la base d'**éléments objectifs** (parfois en allant rechercher dans leurs dossiers les chiffres exacts !). Ce deuxième élément était important pour nous car nous souhaitons aller plus loin qu'une approche binaire de la décision (tout faire, tout faire faire),

- le principal inconvénient a été la lassitude des répondants après quelques questionnaires, ce qui a limité le nombre de projets étudiés.

3.2 Les variables correspondant aux déterminants des choix

Les variables principales directement associées au modèle sont :

- Facteurs stratégiques

• **Importance stratégique du projet**

Cette variable a été mesurée de deux façons : dans un premier temps en demandant au répondant d'évaluer directement l'importance stratégique du projet, dans un deuxième temps en lui demandant d'indiquer quel rôle avait joué à son avis ce critère dans la décision. L'hypothèse correspondante est la suivante : *H1. L'entreprise a davantage tendance à externaliser si l'importance stratégique du projet est faible.*

- Facteurs "industriels"

• **Nouveauté technologique** : le projet met-il en oeuvre des technologies nouvelles pour l'organisation ? Nous avons en effet considéré que la complexité technologique est un critère *relatif* pour l'organisation, qui se mesure surtout en termes de nouveauté pour celle-ci. Une technologie nouvelle est susceptible de déterminer un choix d'externalisation, que l'application soit stratégique (objectif d'apprentissage) ou pas (objectif de limitation des coûts). Cette variable a également été mesurée de deux façons, l'une directe (évaluer le niveau de complexité technologique du projet), l'autre indirecte (quel est à votre avis le poids de la spécificité technique dans la décision). L'hypothèse est la suivante : *H2. L'entreprise a davantage tendance à externaliser si la nouveauté technologique du projet est élevée.*

- **Optimisation charge/capacité.** L'hypothèse est la suivante : *H3. L'entreprise a davantage tendance à externaliser si l'optimisation charge/capacité est un facteur déterminant dans le choix.*

- **Délais.** L'hypothèse est la suivante : *H4. L'entreprise a davantage tendance à externaliser si le délai est un facteur déterminant dans le choix.*

- **Intégration au système d'information.** (à noter que cette variable est également un facteur jouant sur les coûts de transaction). L'hypothèse est la suivante : *H5. L'entreprise a davantage tendance à externaliser si le projet est peu intégré au système d'information existant.*

- Facteurs de contrôle

- **Structure du projet :** le projet était-il au départ peu structuré ou très structuré ? Cette variable est un indicateur de la possibilité d'évaluer le fournisseur au résultat ; elle est rattachée à la difficulté d'évaluation du projet *a priori*. L'hypothèse est la suivante : *H6. L'entreprise a davantage tendance à externaliser si le projet est au départ très structuré.*

- **Spécificité du domaine d'application.** L'hypothèse est la suivante : *H7. L'entreprise a davantage tendance à externaliser si le domaine d'application du projet est peu spécifique.*

3.3 Les données collectées

L'ensemble des variables a été réparti dans deux groupes de questions (voir le questionnaire en annexe). Dans un premier temps, on a demandé aux interlocuteurs d'évaluer un ensemble de 8 variables correspondant à des **éléments descriptifs du projet**. Les variables appartenant à ce groupe sont les suivantes (échelle en cinq points, de 1 faible à 5 fort) :

- 6 variables correspondant aux hypothèses de la recherche : l'importance stratégique, la nouveauté technologique, l'urgence, les liens avec les applications existantes⁵, le degré de structure du projet et la spécificité du domaine d'application.

⁵. A noter que la note 5 (très intégré) a été attribuée systématiquement à tous les projets de type "interfaces" et "maintenance".

- 2 variables complémentaires : la taille, qui est un facteur de contingence "classique" et la perception de la contrainte budgétaire sur le projet.

Le deuxième groupe de variables correspond à une évaluation par les répondants du **poids des différents facteurs dans la décision**. On leur a demandé si, à leur avis, ces facteurs ont influencé le choix entre le recours à la ressource interne, à la régie ou au forfait. On a utilisé une échelle en 5 points (de 1, pas déterminant à 5, facteur déterminant), qui a été ramenée dans le traitement à 3 pour mieux mettre en évidence le profil de réponses. Ce groupe est également composé de 8 variables :

- 5 variables correspondant aux hypothèses de la recherche : l'ajustement charge /capacité, l'importance stratégique, la spécificité technique, les contraintes de délai, l'importance de la connaissance du système d'information actuel,

- 3 variables "globales" qui nous ont été suggérées par les premiers tests du questionnaire :

. l'importance du critère "coût" dans la décision,

. l'importance du critère de contrôle dans la décision. Nous avons voulu voir si les répondants associaient certaines origines de la ressource à la mise en oeuvre d'un meilleur contrôle,

. l'importance du critère d'habitude dans la décision. Cette variable n'est rattachée à aucun de nos modèles théoriques. Nous avons cependant pensé que certains choix pouvaient correspondre en fait à des "non-choix", c'est-à-dire à la reconduction des habitudes de l'organisation en matière de développement, traditionnellement favorables à l'informatique interne.

Section 2 - Analyse descriptive des choix d'externalisation

Le choix d'externalisation est exprimé sur chaque projet par un groupe de 9 variables, correspondant à trois types de solutions (interne, régie et forfait), ce pour trois étapes successives du cycle de vie d'un projet (analyse du besoin, conception, réalisation).

	Analyse du besoin	Conception	Réalisation
interne	ABinter(%)	Concint(%)	Réainter(%)
régie	ABregie(%)	Concregi(%)	Réaregie(%)
forfait	ABforfai(%)	Concforf(%)	Reaforfa(%)

En analysant la structure de ces variables, nous cherchons à répondre à un ensemble d'interrogations :

- Peut-on associer chaque type contractuel de façon privilégiée à une étape donnée du projet, selon notre hypothèse de spécialisation suivant laquelle les caractéristiques respectives des trois étapes (analyse du besoin, conception, réalisation) conduisent à sélectionner pour chacune d'entre elles le mode d'organisation approprié ?

- Y a-t-il au contraire stabilité de l'utilisation d'un mode contractuel tout au long du déroulement d'un projet, quelle que soit l'étape du déroulement ?

- Dans quelle mesure le choix particulier d'un type d'organisation dans les étapes amont (analyse du besoin, conception) influence-t-il celui effectué pour les étapes aval (conception, réalisation) ? Peut-on identifier des schémas de comportement ?

I. Analyse exploratoire : Description des variables résultat (choix d'externalisation)

1.1 Fréquences et moyennes

L'étude des tableaux des fréquences nous amène à formuler trois observations portant respectivement sur les particularités des distributions observées, la fréquence très élevée de la valeur "zéro" (non-utilisation d'une ressource) et les variations de la distribution en fonction des étapes de projet.

1.1.1 Particularités des distributions observées

Notre première observation est de relever la **particularité des distributions observées**, qui sont **très concentrées autour d'une ou deux valeurs très fréquentes**, représentant plus de la moitié de l'effectif total (cas de 7 variables résultat sur 9).

	Valeur la plus fréquente	% des effectifs	2ème valeur la plus fréquente	% des effectifs	% cumulé
A. Besoin interne	100	66.7%	50	13.3%	80.0%
A. Besoin régie	0	83.3%	50	8.3%	91.6%
A. Besoin forfait	0	83.3%	100	6.7%	90.0%
Conception régie	0	51.7%	50	13.3%	65.0%
Conception forfait	0	66.7%	100	11.7%	78.4%
Réalisation régie	0	45.0%	100	16.7%	61.7%
Réalisation forfait	0	56.7%	100	20.0%	76.7%

On note également la fréquence des valeurs extrêmes (0, 100), moyenne (50) ou correspondant à des pourcentages simples (1/4, 1/3). S'agissant de pourcentages d'utilisation des différentes solutions, il est clair que les répondants ont été attirés par des valeurs "repère" plus que par la recherche d'un chiffre précis, d'ailleurs difficile à déterminer. Le critère de référence (budget, temps passés) n'est pas évident et les lacunes du suivi des coûts, en

particulier internes, empêchent une valorisation exacte de la ventilation des ressources employées. Cependant tous nos interlocuteurs nous ont déclaré ne pas avoir de difficulté à identifier un ordre de grandeur correct. En fait, nous avons retenu cette manière d'exprimer l'utilisation des différents choix car elle s'est imposée lors des entretiens préparatoires comme la plus "naturelle" et compréhensible par les interviewés.

Deux variables n'apparaissent pas dans le tableau ci-dessus. La variable "conception interne" est distribuée autour de trois pôles : 100 (23.3%), 50 (20.0%), 0 (18.3%). La variable "réalisation interne" comporte quant à elle une valeur très fréquente - 0 (38.3%)- et une répartition assez équilibrée sur le reste de l'intervalle.

1.1.2 Fréquence élevée de la valeur "zéro" (non-utilisation d'un mode contractuel)

Deuxième élément frappant, c'est la valeur "zéro" (non-utilisation d'un mode contractuel qui est la plus fréquente pour 7 variables sur 9. Les exceptions concernent le pourcentage de l'analyse du besoin et de la conception effectués en interne. Pour l'étape de réalisation, l'ensemble des trois modes a même zéro pour valeur la plus fréquente. Plusieurs remarques peuvent être faites à ce propos :

- Tout d'abord, le non recours à certains types de solutions peut s'expliquer par l'existence d'une **pratique dominante et exclusive** : cela semble être le cas dans l'étape d'analyse du besoin où la ressource informatique est utilisée dans 66.7% des cas comme seul moyen. Cependant, cette explication ne peut être retenue pour les autres étapes.

- Ensuite, il nous faut noter le faible recours au forfait au moins dans les deux premières étapes (moins d'un cas sur cinq pour l'analyse du besoin, un cas sur trois pour la conception, moins d'un cas sur deux en réalisation). La régie, qui n'a pas plus de succès que le forfait dans la première étape, augmente quant à elle très fortement dès la phase de conception (un cas sur deux). De plus faut-il rappeler ici que notre étude, compte tenu de la préférence donnée à la diversité des choix dans la sélection des projets, conduit sans doute à une "sur-représentation" du forfait dans les entreprises étudiées. Il nous paraît également important de signaler que

beaucoup parmi nos interlocuteurs ont assorti leur réponse au questionnaire à l'expression d'une motivation et d'un regret vis-à-vis du mode forfaitaire : motivation pour le développer, regret que cette motivation ne se soit pas encore traduite dans les faits, ce qui nous fait dire qu'à défaut d'être un mode d'organisation utilisé, c'est un mode perçu comme désirable. (Nous reviendrons sur cet aspect plus largement dans le traitement qualitatif des entretiens).

- Dernière remarque sur la fréquence de la valeur zéro, celle-ci correspond au fait que, dans beaucoup de cas, on préfère exclure complètement le recours à un type de solution : à ce stade de l'analyse, nous ne pouvons pas expliquer pourquoi. Différentes hypothèses peuvent cependant être avancées, comme la difficulté de partager un projet en trois sous-ensembles gérés différemment ou encore comme le souci d'éviter d'avoir à gérer trop de relations contractuelles différentes pour la même transaction.

1.1.3. Variation des choix en fonction des étapes du projet

Les choix d'externalisation varient sensiblement entre les trois étapes. Dans une première approche rapide, on peut constater :

- dans l'étape d'**analyse du besoin**, la **dominance du mode interne** utilisée à l'exclusion des deux autres (66.7% des cas).

- dans l'étape de **conception**, cette dominance exclusive s'efface (seulement 23% des cas). Plus généralement, **aucun mode dominant** ne se dégage ; il semble plutôt y avoir **combinaison** de la ressource interne avec l'une ou l'autre des deux autres solutions.

- dans l'étape de **réalisation** on continue de noter cette combinaison des différents modes. **L'informatique interne est à ce stade le mode le moins utilisée** (moyenne 28.2%). De plus, quand elle l'est, c'est quasiment toujours en combinaison avec une autre solution. En revanche, la régie, quand elle est utilisée, est employée comme moyen unique dans 30.3% (10/33) des cas. Ce chiffre monte à 46.2% (12/26) des cas pour le forfait.

Deux questions complémentaires nous préoccupent à ce stade de l'analyse :

Tout d'abord y a-t-il des combinaisons de choix plus fréquentes les unes que les autres au sein d'une même étape ? Nous allons pour répondre à cette question étudier les **solutions mixtes** (couples ou triplets).

Ensuite, comment évoluent les choix au long des étapes identifiées du processus ? En passant d'une étape à l'étape suivante, y a-t-il substitution privilégiée d'un mode d'organisation par un autre ? Pour cela, nous continuerons l'approche descriptive en la focalisant sur l'**étude des corrélations** et sur l'analyse des **évolutions entre les étapes**.

1.2 Les solutions mixtes : combinaisons de modes différents au sein d'une même étape.

Quelles sont les associations les plus fréquentes ? Nous avons récapitulé les réponses obtenues dans le tableau ci-après.

Combinaison de modes d'organisation (nombre de cas sur un effectif de 60)		Analyse du besoin	Conception	Réalisation
Trois modes	Interne / Régie / Forfait	0	3	3
Deux modes	Interne / Régie	10 (48/52)*	22 (51/49)*	19 (39/61)*
	Interne / Forfait	6 (47/53)*	10 (56/43)*	10 (38/62)*
	Régie / Forfait	0	0	1
Mode unique	Interne	40	14	5
	Régie	0	4	10
	Forfait	4	7	12

* les chiffres indiqués entre parenthèses indiquent la répartition moyenne entre les deux formes.

Tableau 8.3 Les solutions mixtes :

combinaison de différents modes d'organisation au sein d'une même étape de projet

Pour l'étape d'analyse du besoin, on retrouve la dominance de l'informatique interne précédemment identifiée, seule ou en association avec la régie ou le forfait. Dans 4 cas, il a été retenu d'externaliser complètement le projet dès cette phase initiale. Il est intéressant de noter

que l'on a alors choisi la méthode du forfait, qui revient à transférer la maîtrise du projet à l'extérieur.

Pour l'étape de conception, c'est l'association **interne/régie** est la plus fréquente ; le *tout interne* décroît très fortement et le mixte "interne/forfait" se confirme. La répartition moyenne entre les modes d'organisation combinés semble équilibrée. Il faut noter cependant que cette répartition moyenne correspond certes à certains projets ayant une répartition 50/50, mais que l'on constate également des répartitions déséquilibrées.

Pour l'étape de réalisation, on retrouve l'association **interne/régie** et, à un moindre degré, l'association **interne/forfait**. Dans ces combinaisons, la ressource interne est moins utilisée que la ressource externe. La **régie** complète et le **forfait** complet apparaissent de plus comme des solutions spécifiques à cette phase.

Il faut également noter la rareté du choix simultané des trois modes d'organisation et l'opposition entre le mode régie et le mode forfait, très rarement associés. Cette opposition explique peut-être la rareté des solutions triples.

1.3 Les variations entre les étapes.

On constate des variations importantes du pourcentage d'utilisation des modes d'organisation *entre les différentes étapes d'un même projet*. Les graphes ci-après illustrent, pour chaque type de solution, cette variation entre les étapes prises deux à deux. Chaque graphe reprend les 60 projets étudiés (Fig. 8.1).

Pour réaliser ces graphes, nous avons calculé à chaque fois la différence du taux d'utilisation des trois modes d'organisation entre deux étapes d'un même projet. Les 60 projets forment l'axe des abscisses, la variation du taux d'utilisation est représentée en ordonnée. Par exemple, sur le projet n°2, entre l'analyse du besoin et la conception, il y a une chute du taux d'utilisation de l'informatique interne de 80%, compensée par une augmentation de 20% de la régie et de 60% du forfait.

L'intérêt de ces graphes est de montrer quel mode d'organisation se substitue à un autre au fur et à mesure de l'avancement du projet.

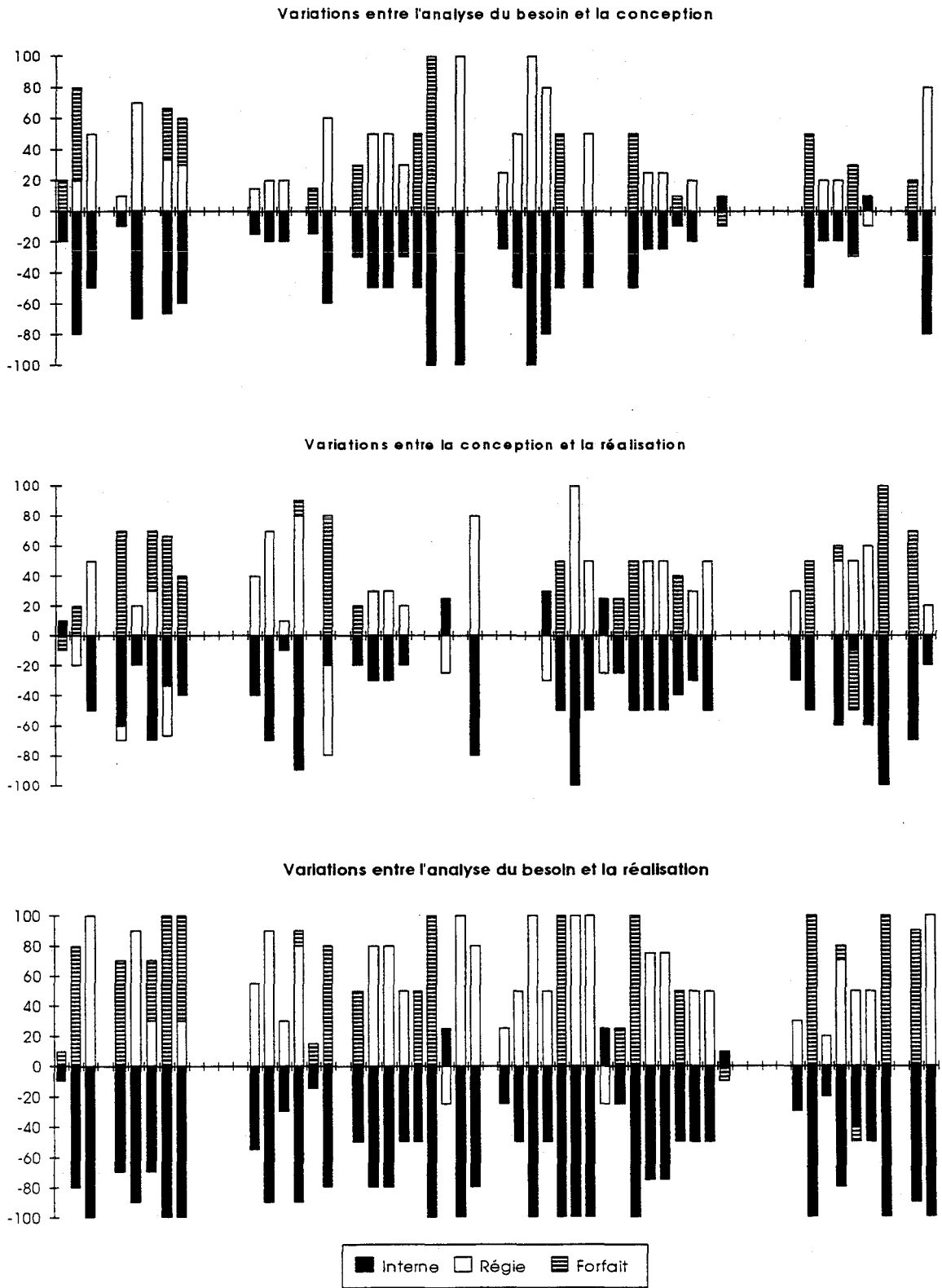


Figure 8.1. Evolution des choix d'externalisation entre les différentes étapes, sur les 60 projets

Ces différents résultats montrent :

– une forte décroissance de l'utilisation de la ressource interne au fur et à mesure que l'on avance dans le projet. Une première baisse, très accentuée, se produit lors du passage de l'analyse du besoin à la phase de conception (36 projets concernés); une seconde baisse, également très nette, accompagne le passage en phase de réalisation. Au total, l'utilisation de la ressource interne décroît (en pourcentage) de l'étape d'analyse du besoin à la réalisation dans 45 projets et n'augmente que dans 3 cas.

– en revanche, l'utilisation des modes régie et forfait croît au fur et à mesure de l'avancement des étapes du projet. L'accroissement se fait pour chacun des deux modes en deux hausses équilibrées (passages de l'analyse du besoin à la conception et de la conception à la réalisation).

- dans la quasi-totalité des cas, il n'y a pas de changement du mode contractuel de pilotage du fournisseur extérieur au cours du projet (pas de passage de la régie au forfait et pas de passage du forfait à la régie sur un même projet).

Sur l'ensemble des projets, le taux d'utilisation moyen des différents modes d'organisation varie de la façon suivante (fig. 8.2).

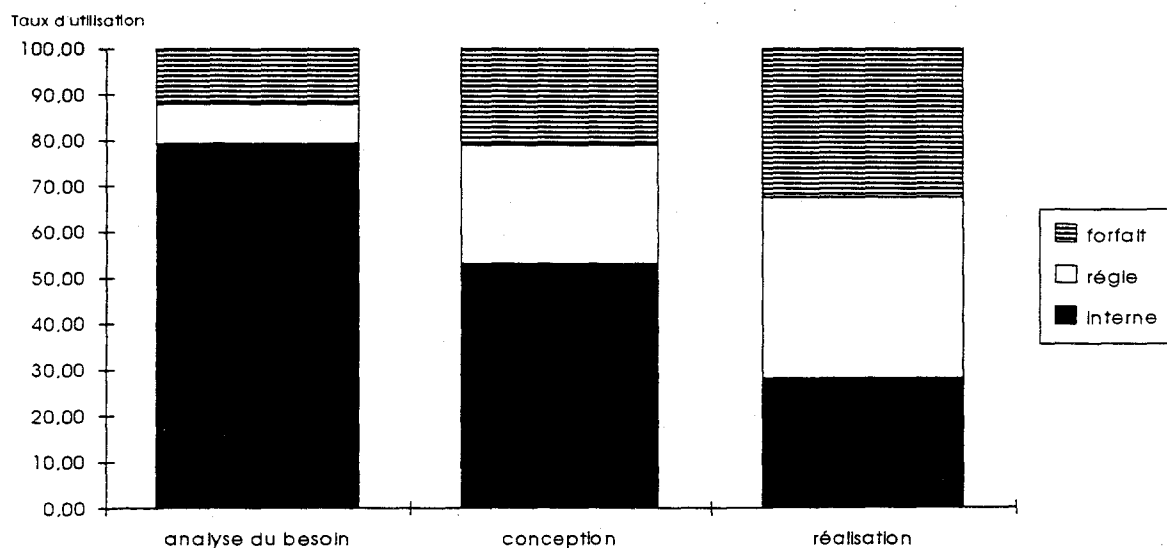


Figure 8.2. Taux d'utilisation moyen des trois modes d'organisation dans les différentes étapes du processus

On constate ici le maintien en interne de l'analyse du besoin, une externalisation relativement forte en phase de conception et très accentuée en phase de réalisation.

Ces résultats sont cohérents avec notre hypothèse d'une différenciation des choix d'externalisation en fonction des étapes du processus de développement.

1.4 Etude des corrélations

Nous avons voulu savoir si le lien entre les choix d'externalisation aux différentes étapes est significatif. La matrice des corrélations est reprise dans le tableau 8.4 (60 individus).

	Analyse du besoin			Conception			Réalisation		
	interne	régie	forfait	interne	régie	forfait	interne	régie	forfait
A. Besoin interne	1.0000								
A. Besoin régie	-.4824 (.000)	1.0000							
A. Besoin forfait	-.7816 (.000)	-.1693 (.196)	1.0000						
Conception interne	.6281 (.000)	-.3044 (.018)	-.4900 (.000)	1.0000					
Conception régie	-.0929 (.480)	.5947 (.000)	-.3189 (.013)	-.5077 (.000)	1.0000				
Conception forfait	-.5718 (.000)	-.2494 (.055)	.8209 (.000)	-.5649 (.000)	-.4240 (.001)	1.0000			
Réalisation interne	.2524 (.052)	-.0559 (.671)	-.2442 (.060)	.6056 (.000)	-.2881 (.026)	-.3605 (.005)	1.0000		
Réalisation régie	.1065 (.418)	.3746 (.003)	-.3865 (.002)	-.1153 (.380)	.6967 (.000)	-.5456 (.000)	-.3514 (.006)	1.0000	
Réalisation forfait	-.2972 (.021)	-.3175 (.013)	.5605 (.000)	-.3561 (.005)	-.4485 (.000)	.8033 (.000)	-.4331 (.001)	-.6917 (.000)	1.0000

Tableau 8.4. Matrice des corrélations entre les choix d'externalisation

Les 3 variables résultat au sein d'une même étape sont par construction liées (total des trois modes = 100%). Le seul point remarquable est l'absence de lien significatif entre les modes "régie" et "forfait" dans l'étape analyse du besoin : ceci s'explique par la dominance du mode "interne" dans cette étape. Les corrélations entre les étapes séquentielles du projet permettent quant à elles d'établir la non-indépendance des choix contractuels entre les différentes étapes d'un projet. Les figures de la page suivante reprennent les résultats observés.

Les corrélations entre les choix d'externalisation aux différentes étapes des projets

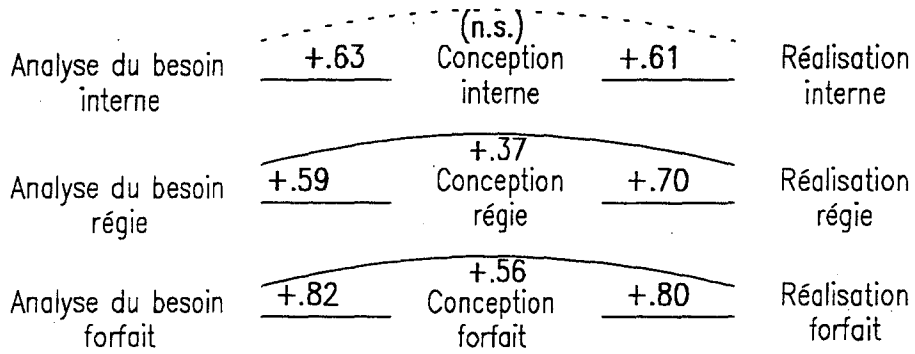


Figure 8.3. Les corrélations entre l'utilisation d'une même solution tout au long du projet

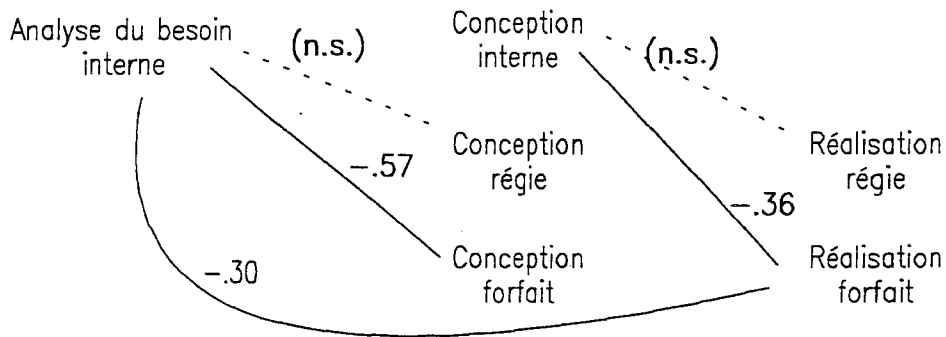


Figure 8.4. Les corrélations interne / régie et interne / forfait

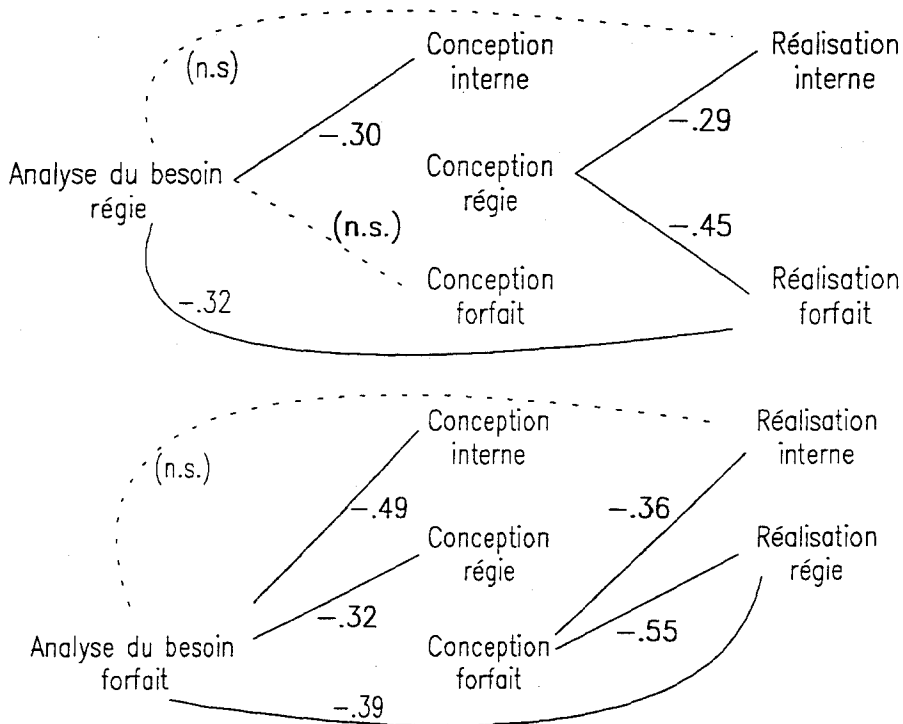


Figure 8.5. Les corrélations régie / autres modes et forfait / autres modes

1.4.1 La non-indépendance des choix

La figure 8.3 reprend les corrélations entre l'utilisation d'un même mode contractuel (interne, régie, forfait) dans les différentes étapes du projet. Ces corrélations sont significatives et fortement positives dans tous les cas étudiés, sauf pour la relation (analyse du besoin interne, réalisation interne). **Les choix d'externalisation effectués aux différentes étapes de projet ne sont pas indépendants.** L'utilisation d'un mode contractuel donné en début de projet tend à se prolonger dans les étapes ultérieures.

La seule exception à cette règle est la corrélation (analyse du besoin en interne, réalisation en interne). Nous avons pu voir qu'alors que dans la première étape étudiée la ressource interne est fortement dominante, celle-ci tend à décroître dans les deux étapes suivantes. **Démarrer un projet en interne n'implique donc plus aujourd'hui, d'après nos données, de le réaliser en interne.**

La dépendance des choix est en revanche très accentuée dans le mode forfaitaire (un projet engagé au forfait est réalisé au forfait) et, pour tous les modes, lors du passage de la phase de conception à celle de réalisation. On aurait pu penser qu'un "contrat de réalisation", élaboré en fin de phase de conception provoquerait une modification nette du mode contractuel et le recours à une gestion par le marché (évaluation au résultat) pour la réalisation. Nous nous attendions donc à un basculement au niveau de la phase de réalisation, qui n'est pas confirmé par nos résultats.

1.4.2 Les relations entre la ressource interne et les autres modes d'organisation

La figure 8.4 montre l'un des résultats à notre avis les plus intéressants de cette étude. On observe une relation très différente entre le mode interne et la régie d'une part et le forfait d'autre part.

La relation interne/régie est neutre (non significative). Nous interprétons ce résultat comme l'indication d'une **bonne possibilité de substitution de la ressource interne par la régie externe.** Un contrat en régie peut s'insérer naturellement dans le processus de projet en phase

de conception ou en phase de réalisation. Les pratiques observées corroborent l'une des toutes premières remarques faites au début de ce travail (chapitre 1) : nous nous étions alors étonnés de constater le peu d'études spécifiques réalisées sur l'externalisation dans le domaine des projets de développement applicatifs et l'habitude prise par les praticiens et les chercheurs de confondre ressource interne et ressource externe dans une même approche. Tant que la régie est le mode d'externalisation partielle⁶ retenu, cette assimilation est confirmée par les résultats obtenus. Ce n'est plus le cas avec une externalisation au forfait.

La relation **interne/forfait** est en effet appuyée sur l'**opposition**. Une forte implication de la ressource interne à une phase du projet est corrélée négativement avec l'utilisation du mode forfaitaire dans une phase ultérieure. La substitution, en fonction des étapes de l'avancement du projet, d'une ressource interne par un forfait n'est manifestement pas habituelle. D'un autre côté, engager un projet au forfait dès la phase d'analyse du besoin est difficile, et de plus à notre avis rarement souhaitable, compte tenu de la forte spécificité des activités à ce stade et des enjeux pour l'organisation. Si, comme nous le pensons, le mode forfaitaire est en règle générale la solution la plus efficiente pour la phase de réalisation, alors il est nécessaire que les entreprises changent leurs habitudes de gestion de projet et développent des méthodes leur permettant une gestion différenciée selon les étapes du développement.

1.4.3 Les relations régie / autres modes et forfait / autres modes

La figure 8.5 complète certaines observations précédentes.

Tout d'abord, on retrouve sous une autre forme le phénomène de dépendance des choix. L'emploi d'un mode contractuel à une étape est corrélé positivement avec l'emploi de ce même mode à l'étape suivante et négativement avec l'utilisation des autres modes contractuels.

Nous avons constaté une opposition nette entre le mode interne et le mode forfaitaire. Cette opposition ne se retrouve pas dans tout à fait dans les mêmes termes entre la régie et le forfait. La corrélation (analyse du besoin régie, conception forfait) n'est pas significative.

⁶. Rappelons que nous interprétons la régie comme une forme de gouvernance hybride (régime d'incitations proche du marché mais contrôle de type hiérarchique), qui correspond donc à une externalisation partielle.

L'analyse détaillée des cas montre qu'en fait ce ne sont pas les mêmes projets qui sont concernés. De la même façon, seulement 3 projets sur les 60 passent de la régie au forfait entre la phase de conception et de réalisation. On peut ainsi conclure que quand la relation contractuelle avec le fournisseur est engagée en amont du projet sur le principe d'une maîtrise interne et d'un contrôle du comportement, cette relation n'est *pas modifiée* en phase de réalisation dans le sens d'une délégation et d'un contrôle du résultat.

Enfin, le mode forfait confirme son opposition aux deux autres modes.

Nous allons maintenant continuer l'analyse descriptive des variables résultat en utilisant deux méthodes statistiques différentes. Tout d'abord, nous avons réalisé une analyse en composantes principales à partir des neuf variables de résultat. L'objectif est d'obtenir une visualisation globale des projets et de mettre en évidence les variables orthogonales latentes. Ensuite, nous avons établi une typologie des projets en 6 classes en utilisant la méthode de Ward appliquée aux coordonnées des points sur les 3 premiers axes de l'analyse en composantes principales. Les traitements ont été réalisés à l'aide du logiciel SPSS.

II. Analyse en composantes principales des 9 variables résultat (choix d'externalisation).

L'analyse en composantes principales est une méthode statistique descriptive adaptée au cas de variables quantitatives. On trouvera une présentation mathématique de la méthode ainsi que l'exposé de ses principaux avantages dans le précieux ouvrage de M. Tenenhaus (1994). Rappelons que l'analyse en composantes principales permet d'obtenir une carte des individus (projets de développement) en fonction de leur proximité et une carte des variables en fonction de leurs corrélations (cercle des corrélations).

L'analyse en composantes principales consiste à rechercher un petit nombre de nouvelles variables, les composantes principales, non corrélées entre elles et résumant aussi bien que possible les données de départ.

L'analyse en composantes principales sur nos 9 variables résultat donne les résultats suivants :

Composante principale	Valeur propre	Pourcentage de variance expliquée	Pourcentage cumulé
1	3.87897	43.10	43.10
2	2.82532	31.39	74.49
3	1.16378	12.93	87.42
4	.57695	6.41	93.83
5	.41117	4.57	98.40
6	.14381	1.60	100.00
7	.00000	.00	100.00

Nous ne retenons que les trois premières valeurs propres pour deux raisons. Tout d'abord, nous avons respecté la règle généralement admise qui est de ne retenir que les valeurs propres dont la valeur est supérieure à 1. Ensuite, le pourcentage de variance expliquée décroît fortement dès la quatrième valeur propre (6.4%). On aboutit avec les trois premières valeurs propres à un pourcentage total de variance expliquée de 87.4%, ce qui peut être considéré comme satisfaisant.

2.1 Interprétation des axes.

Pour interpréter les différents axes, nous utilisons la matrice des corrélations entre les variables d'origine et les composantes principales.

Matrice des corrélations	Première composante principale	Seconde composante principale	Troisième composante principale
ABINTER	.64557	-.53302	-.50659
ABREGIE	.21793	.71284	.50979
ABFORFAIT	-.88152	.09214	.20698
CONCINT	.54877	-.74675	.11725
CONCREGI	.40662	.83521	-.08242
CONCFORF	-.96625	-.01467	-.04375
REAINTER	.38047	-.56471	.66679
REAREGIE	.59087	.64493	-.26174
REAFORFA	-.86233	-.18527	-.26237

Tableau 8.5. Corrélations entre les variables d'origine et les composantes principales

Le **premier axe** (première composante principale) oppose du côté négatif le mode forfait sur l'ensemble des étapes (conception, analyse du besoin et réalisation) à toutes les autres variables qui sont, elles, corrélées positivement.

En termes de mode de pilotage contractuel et de gestion de projet, on trouve donc à gauche une évaluation par le résultat et à droite une évaluation continue du comportement.

Nous interprétons le premier axe comme un axe contractuel qui oppose d'une part, à gauche, les contrats basés sur une organisation par le marché (forfait) et d'autre part, à droite, les contrats basés sur une organisation hiérarchique ou hybride (interne, régie).

Le **deuxième axe** est corrélé positivement aux variables conception en régie, analyse du besoin en régie et réalisation en régie. Il est corrélé négativement aux variables conception interne, réalisation interne et analyse du besoin interne. Il vient donc détailler le premier axe en dissociant les modes interne et régie.

Le deuxième axe dissocie les contrats interne (-) et régie (+). Le premier plan principal permet donc de visualiser les 3 modes contractuels étudiés.

Cette configuration apparaît parfaitement quand on représente le cercle des corrélations sur le premier plan principal.

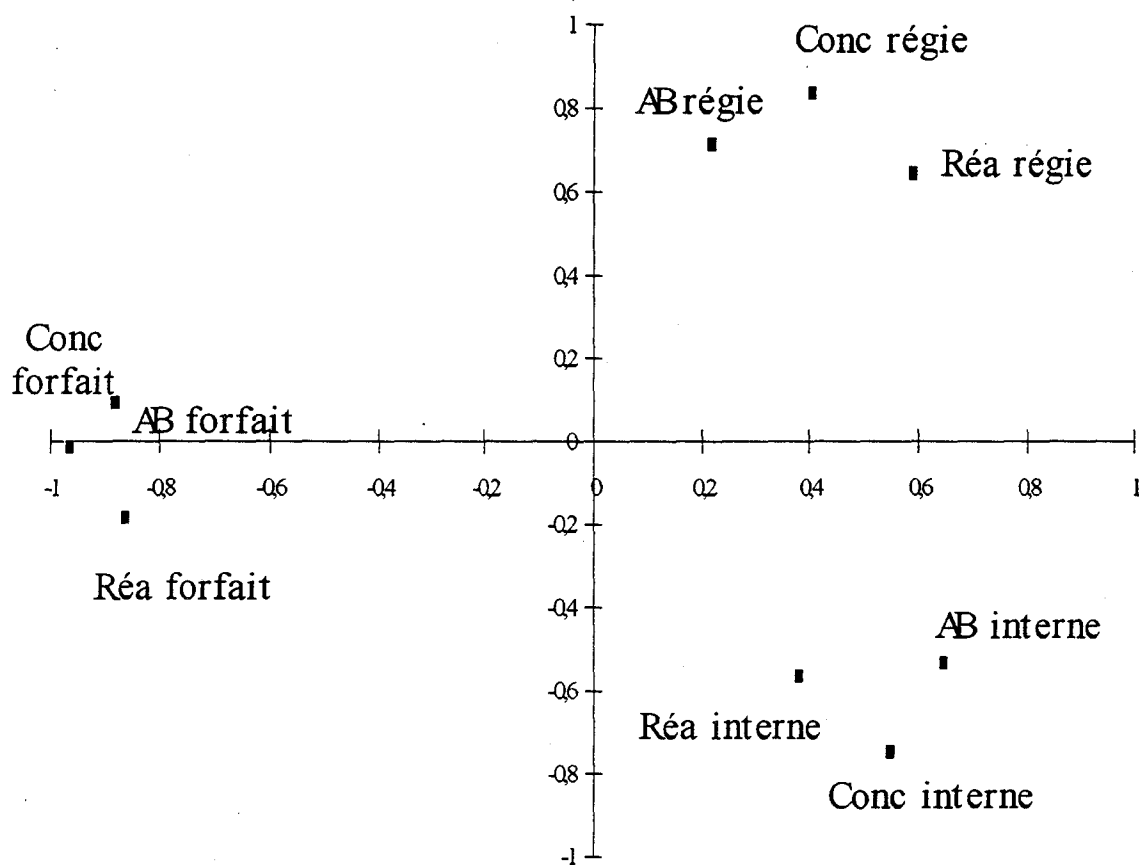


Figure 8.6. Cercle des corrélations sur le premier plan principal

La figure 8.7 montre la représentation des 60 projets dans ce premier plan principal.

On obtient un graphique avec une forme en "V" couché caractérisée par une étonnante continuité (à première vue, il ne se dégage pas de rupture très nette dans la courbe). Le graphe ne fait pas apparaître de points isolés et atypiques ; il est très fortement influencé par les projets au forfait situés à gauche de l'axe 1.

Le troisième axe oppose la réalisation interne (corrélation positive) à l'analyse du besoin en interne (corrélation négative)⁷. On aura donc d'un côté les projets pour lesquels l'informatique interne a été fortement utilisée en phase de programmation et de l'autre ceux pour lesquels l'intervention interne s'est principalement réduite à la définition initiale du projet. Cette distinction nous renvoie directement aux différents rôles du service informatique. Les experts, attachés à leur rôle technique, assurent en interne la phase de réalisation. Les "managériaux" sont davantage centrés sur l'utilisation de la technologie *appliquée aux problèmes de l'organisation* et acceptent de déléguer leur rôle technique. Ils sont très présents en phase amont (et sans doute aval, mais notre étude ne permet pas de le démontrer puisque nous avons exclu la phase d'installation de la recherche terrain).

Nous interprétons l'axe 3 comme un axe de **positionnement du rôle de l'informatique interne dans le processus de développement** : il oppose les "réalisateurs", très impliqués dans la phase de programmation (côté positif de l'axe) aux "instigateurs" qui initient les projets, établissent les liens avec les utilisateurs et délèguent ensuite la réalisation (côté négatif de l'axe).

L'axe 3 permet donc d'établir une distinction particulièrement intéressante et très nouvelle par rapport au premier plan principal. On peut en utilisant cet axe composer deux autres plans principaux.

⁷ . L'analyse du besoin en régie est également fortement corrélée positivement à cet axe ; cet effet est mécanique car le total de la phase analyse du besoin est fixé à 100 par construction et la régie est le mode de substitution par rapport à l'informatique interne. Nous interprétons donc l'axe à partir des deux fortes corrélations sur l'utilisation du mode interne.

Le **second plan principal** (axe 1 ; axe 3) isole, comme le premier plan, les projets réalisés au forfait à gauche et à proximité de l'axe horizontal (cf. fig. 6, cercle des corrélations). A droite se trouvent les projets gérés en interne et en régie. Les projets dont la réalisation (programmation) a été faite en interne sont situés en haut du graphique, les autres dans la partie inférieure.

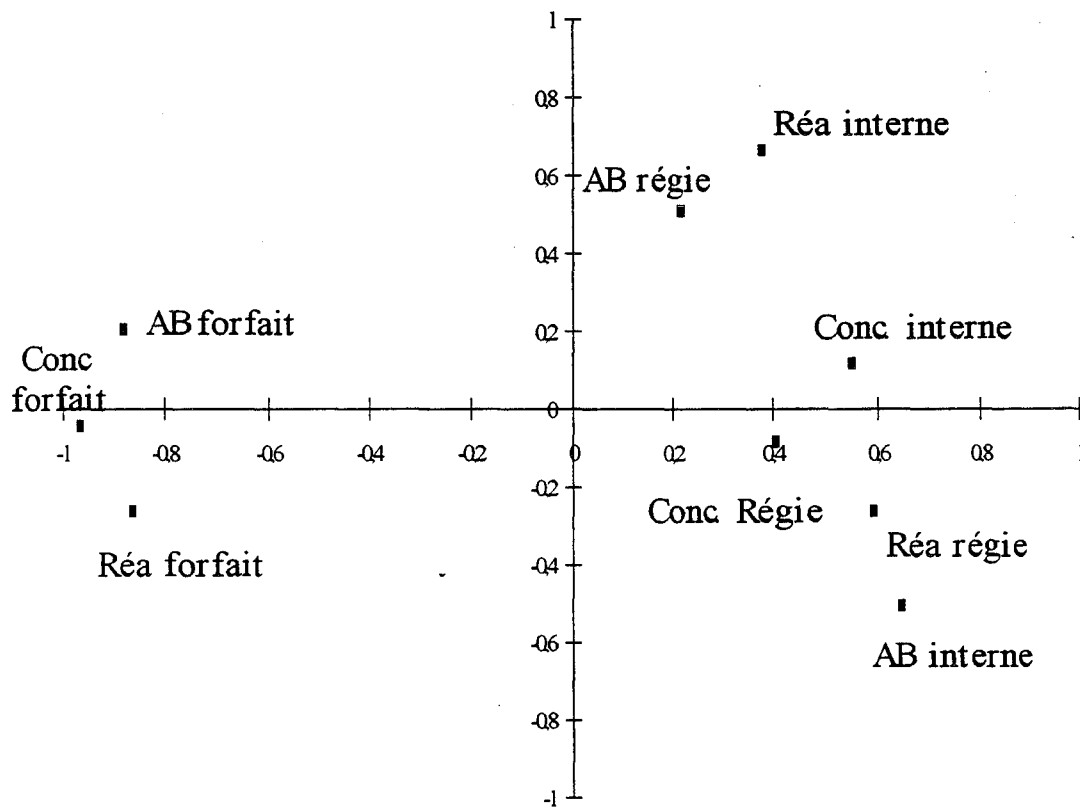


Figure 8.8. Cercle des corrélations sur le second plan principal (axe 1 ; axe 3)

La figure 8.9 montre la représentation des projets sur ce deuxième plan principal.

Analyse en composantes principales

(second plan principal)

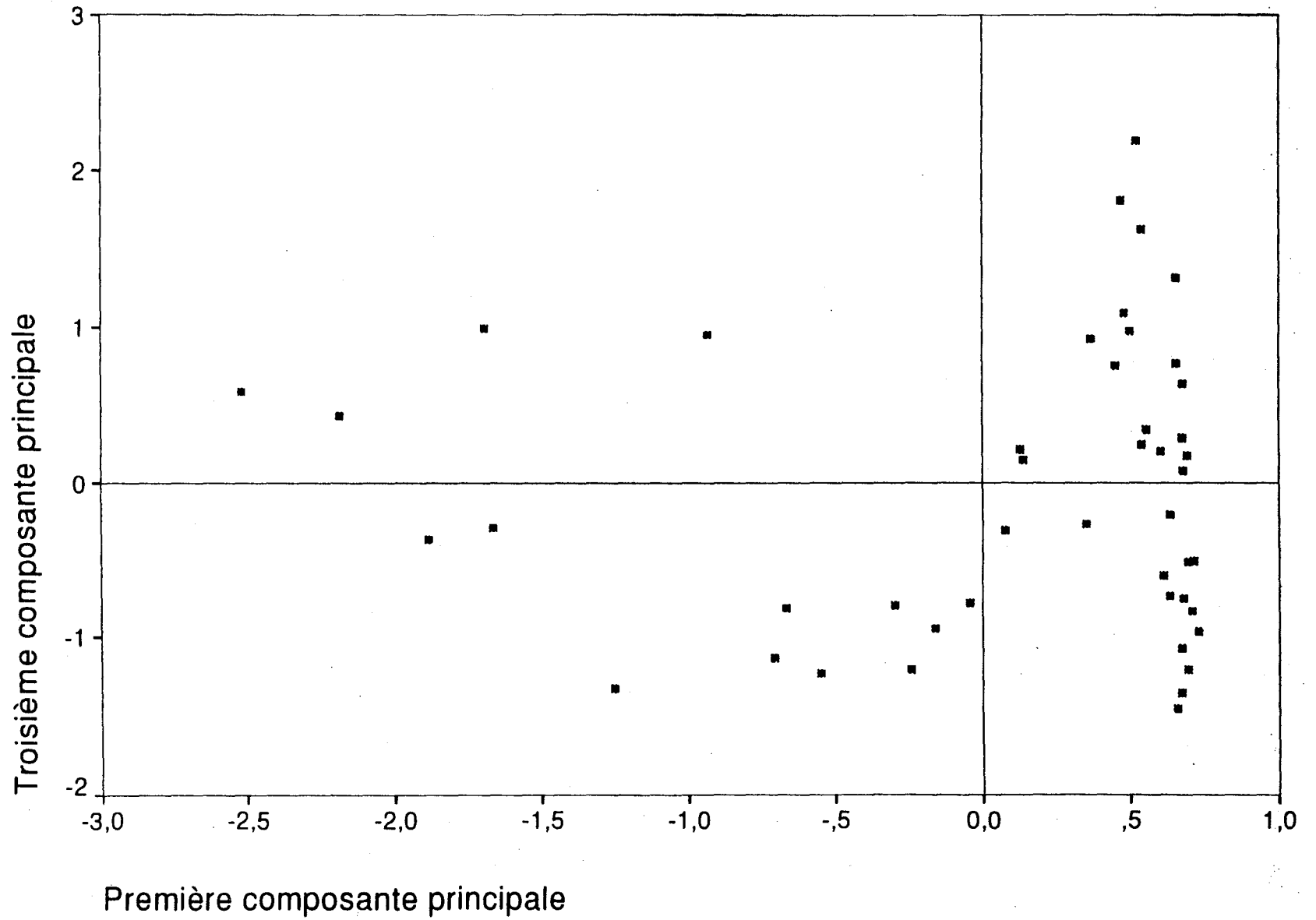


Figure 8.9. Représentation des projets sur le second plan principal

La représentation du second plan principal permet de différencier les projets au forfait selon la part qui a été laissée à la réalisation interne, soit faible (projets situés en dessous de l'axe horizontal), soit forte (projets situés au-dessus de l'axe). Cependant, l'interprétation de la partie droite du schéma reste relativement confuse, car ce plan ne permet pas de séparer clairement les modes interne et régie; différenciation que nous allons en revanche trouver dans le **troisième plan principal**. Ce dernier plan (axe 2 ; axe 3) regroupe au centre les projets au forfait mais permet une interprétation des projets menés en interne et en régie très intéressante, à partir d'un découpage du plan en quatre cadrans :

- du côté gauche du plan figurent les projets à dominance interne : en haut se trouvent les projets de "réalisation interne" et en bas ceux pour lesquels le poids de la ressource interne a été le plus important en phase amont ("initiateurs").

- du côté droit du plan figurent les projets en régie : en bas les "régies complètes de réalisation" et en haut les projets où l'on a pratiqué une utilisation conjointe de la régie et de la ressource interne en phase de réalisation.

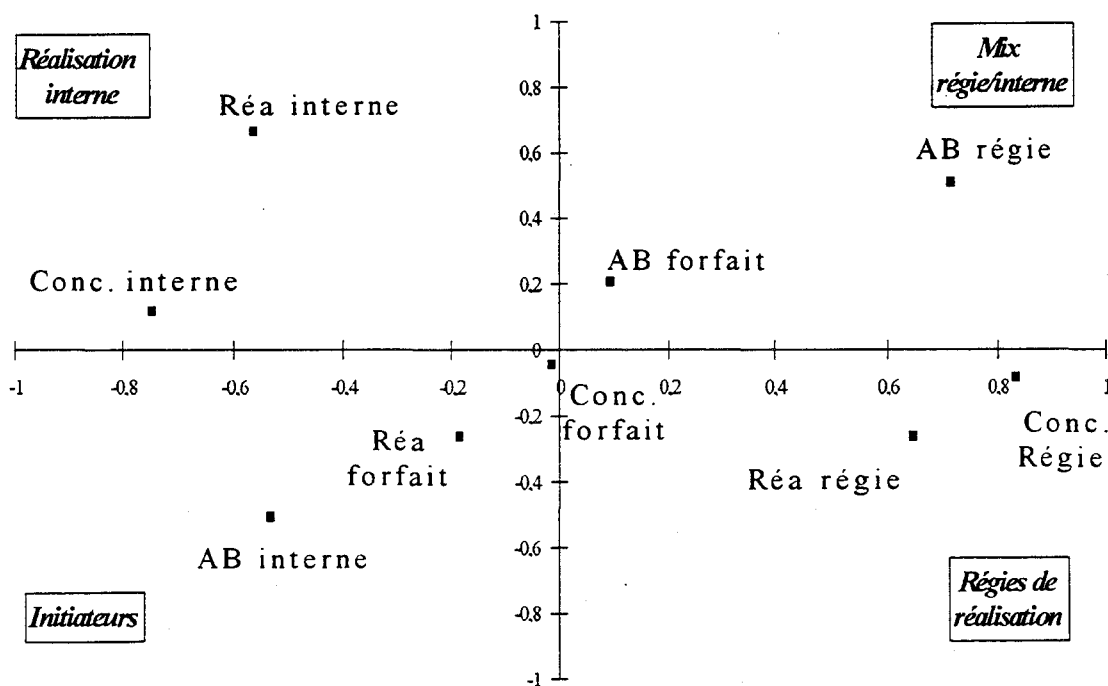


Figure 8.10. Cercle des corrélations sur le troisième plan principal (axe 2 ; axe 3)

Analyse en composantes principales

(troisième plan principal)

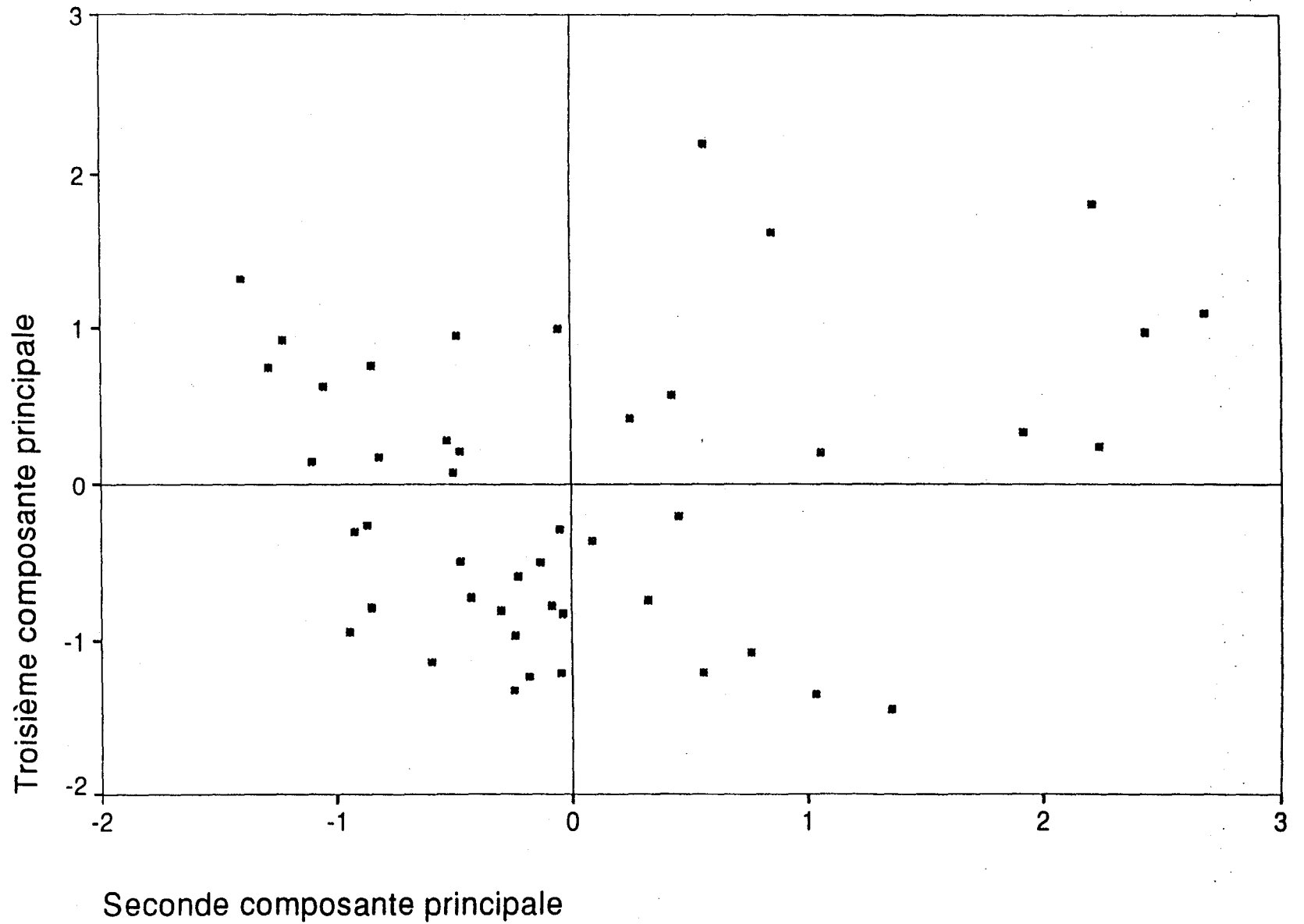


Figure 8.11. Représentation des projets sur le troisième plan principal

La figure 8.11 montre la répartition des projets sur le troisième plan principal. Les quatre cadrans du plan sont occupés, ce qui illustre d'une autre manière la bonne possibilité de substitution déjà évoquée entre la régie et la ressource interne aux différentes étapes du développement.

Pour approfondir l'étude descriptive de nos variables résultat, nous allons maintenant compléter l'approche de l'analyse en composantes principales par l'élaboration d'une typologie, que nous chercherons à visualiser sur les différents plans principaux.

III. Elaboration d'une typologie de projets

Nous utilisons pour élaborer cette typologie de projets une classification hiérarchique basée sur le critère de Ward, appliquée aux coordonnées des points sur les trois premières composantes principales. Les calculs et les graphes ont été effectués à l'aide du logiciel SPSS.

La qualité de la typologie est mesurée par le rapport inertie inter-classes sur inertie totale (soit la part d'inertie expliquée par le modèle), que nous retrouvons dans le tableau suivant :

Nombre de classes	Inertie expliquée	Composition des groupes (effectif)					
		Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
4 classes	71,84%	9	33	9	9		
5 classes	79,30%	9	24	9	9	9	
6 classes	84,85%	9	10	9	9	9	14

Nous avons analysé les trois typologies et choisi de retenir la classification en six groupes, qui est la plus interprétable et permet de plus d'obtenir des effectifs de groupes relativement équilibrés. La part d'inertie expliquée est élevée (84,85%). Il faut cependant signaler que les deux groupes supplémentaires qui apparaissent en faisant passer la typologie de quatre à six groupes sont assez proches : le groupe de 33 individus est d'abord éclaté dans la typologie à 5

classes en deux groupes de 9 et 24, ce dernier étant lui-même éclaté en deux groupes de 10 et 14 dans la typologie à 6 classes. Au final, les groupes 2, 5 et 6, tous issus de la classe 2 dans la typologie à 4 classes sont relativement plus proches les uns des autres que les autres groupes.

Pour interpréter cette typologie nous l'avons représentée sur les différents plans principaux.

Le **premier plan principal** (figure 8.12) permet une visualisation assez claire des différents groupes et fournit les premiers éléments d'appréciation :

- les groupes **1** et **2** sont composés de projets pour lesquels on a utilisé le **forfait** (davantage de forfait dans le groupe 1 que dans le groupe 2).

- les groupes **3** et **5** sont composés de projets pour lesquels on a utilisé de la **régie**.

- les groupes **4** et **6** sont composés de projets pour lesquels on a surtout fait appel à **l'informatique interne**.

Représentation de la typologie sur le premier plan principal

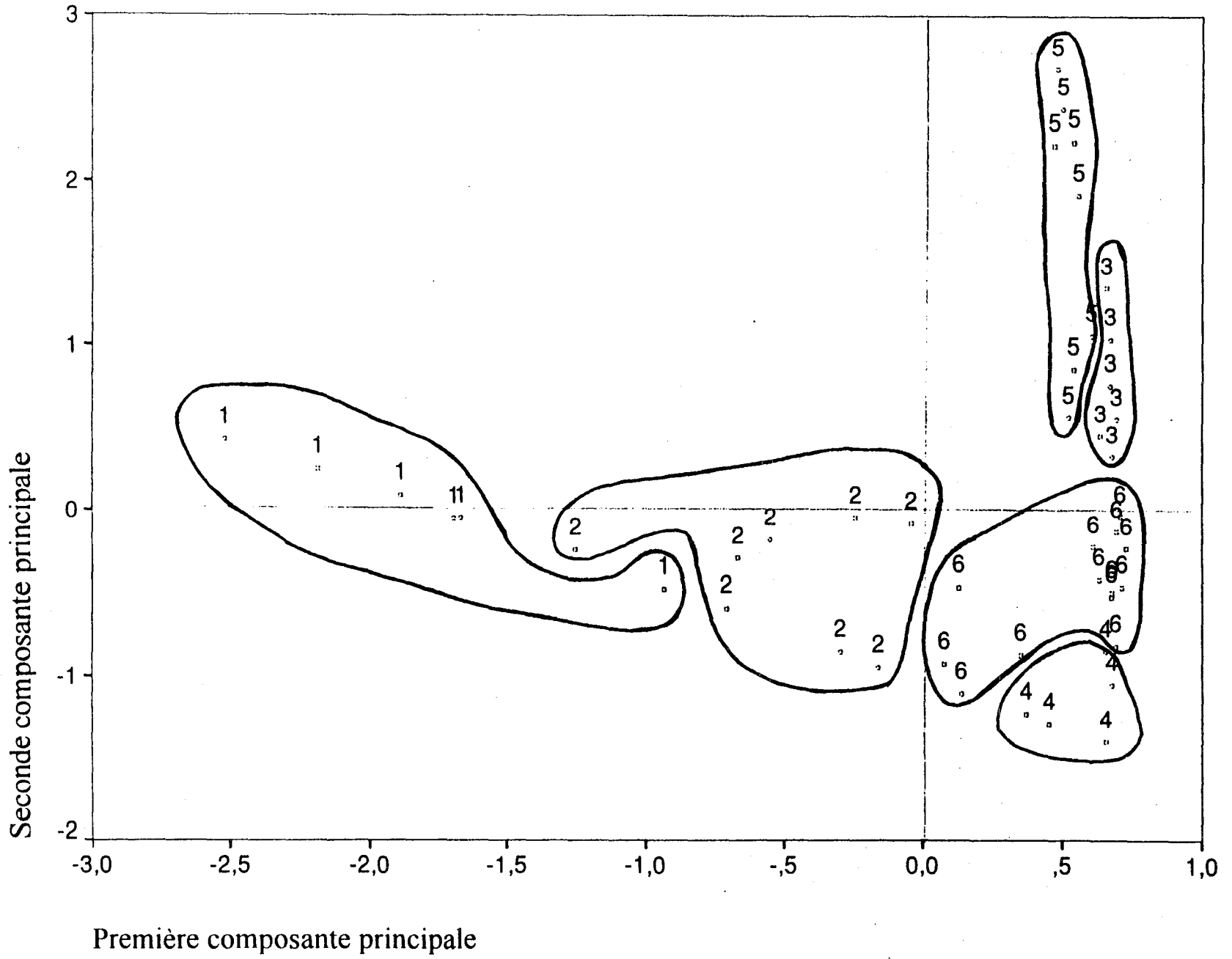


Figure 8. 12. Représentation de la typologie sur le premier plan principal

3.1 L'interprétation des deux groupes de projets utilisant le forfait (groupes 1 et 2)

Le **second plan principal** éclaire la séparation en deux groupes des projets au forfait (figure 8.13) :

- le **groupe 1** (situé plutôt au-dessus de l'axe horizontal) correspond aux projets pour lesquels le forfait a été utilisé dès la phase d'analyse du besoin.

- le **groupe 2** (situé nettement au-dessous de l'axe horizontal) correspond en revanche à des projets où l'analyse du besoin a été faite en interne et la réalisation externalisée au forfait : ce sont les "forfaits de réalisation".

Représentation de la typologie sur le second plan principal

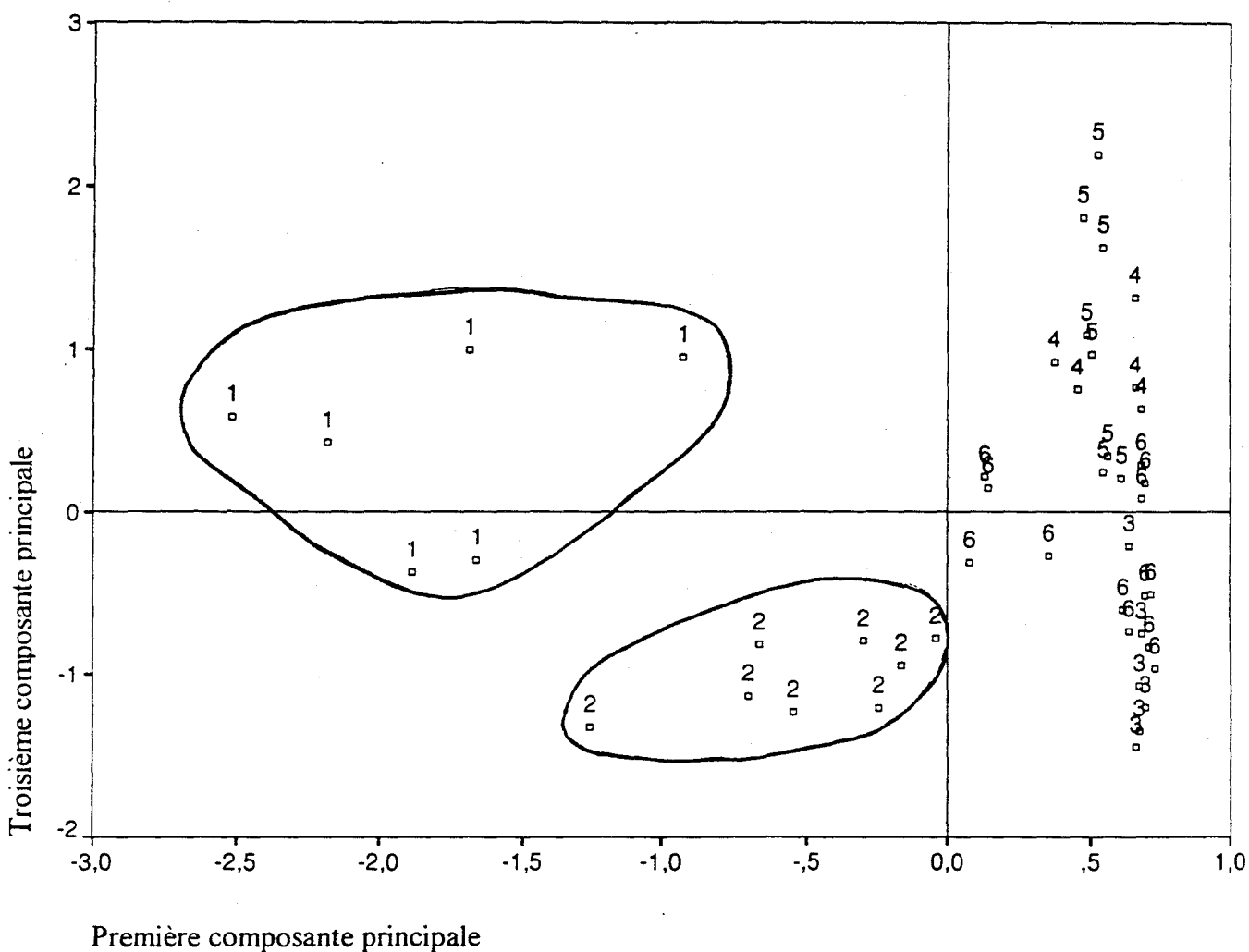


Figure 8.13. Représentation de la typologie sur le second plan principal (axe 1 ; axe 3)

Pour préciser l'interprétation, nous avons également repris le détail des projets concernés dans le tableau 8.6.

groupe 1 - les "forfaits de tout ou partie du projet"								
Numéro du projet	Analyse du besoin		Conception			Réalisation		
	% interne	% forfait	% interne	% forfait		% interne	% forfait	
1	20	80		100		10	90	
11	50	50	50	50		50	50	
20		100		100			100	
21	50	50	20	80			100	
25	50	50		100			100	
30		100		100			100	
46	20	80	30	70		30	70	
47		100		100			100	
48		100		100			100	
groupe 2 - les "forfaits de réalisation"								
Numéro de projet	Analyse du besoin		Conception			Réalisation		
	% interne	% forfait	% interne	% régie	% forfait	% interne	% régie	% forfait
2	100		20	20	60	20		80
8	100		33	33	33			100
9	100		40	30	30		30	70
19	100		40	60		20		80
26	100				100			100
35	100		50		50			100
40	100		50		50			100
52	100		50		50			100
57	100		100					100
59	100		80		20	10		90

Tableau 8.6. Les deux logiques d'externalisation au forfait : détail des projets

A partir de ces différents éléments, notre interprétation est que **les deux groupes au forfait font apparaître deux logiques d'externalisation très différentes :**

- dans le premier cas, *on a identifié dès l'origine un sous-ensemble du projet susceptible d'être entièrement externalisé*. C'est ce sous-ensemble qui est externalisé complètement, quelle que soit l'étape du projet. **On est dans une logique d'externalisation par projet.**

- dans le second cas, on externalise *en fonction des caractéristiques des différentes étapes du projet*. **On est dans une logique d'externalisation de certaines activités** (différenciées selon les étapes) au sein du processus de développement.

3.1.1 Premier type de forfait : la logique d'externalisation de projet

Dans ce premier groupe, l'objectif du forfait est d'externaliser totalement une partie du développement, y compris dans la phase amont d'analyse du besoin. Cela correspond à un véritable **transfert de la responsabilité de tout ou partie du projet chez le fournisseur**, l'informatique interne ne jouant qu'un rôle de vérification du résultat à la réception de l'application. Cette situation est nouvelle et il nous paraît intéressant de détailler les différents cas. Sur les 9 projets de ce groupe :

- 4 correspondent à des travaux complémentaires effectués autour de progiciels spécialisés sur un métier et ont été confiés à un spécialiste du produit (le vendeur ou une SSII ayant des références d'installation du progiciel). La "compétence produit" a conduit à déléguer y compris la phase d'analyse du besoin ; de plus il s'agit ici de projets mettant en oeuvre des technologies nouvelles pour l'organisation. Dans deux cas, les travaux complémentaires au forfait faisaient partie du contrat d'origine au côté du progiciel et composaient ainsi la "solution complète" achetée par le client.

- 2 sont des projets de maintenance évolutive ; l'un correspond en fait à un réaménagement très important de l'application existante, sur de nouvelles bases techniques.

- 3 sont des nouveaux développements. Les interlocuteurs concernés ont évoqué le caractère très technique du projet et un problème de capacité des équipes internes. Dans un cas, le motif évoqué était également politique : "l'utilisateur voulait absolument une obligation de résultat forte et ne faisait pas confiance à l'informatique interne". Face à cette réaction, le

responsable de l'informatique interne a organisé un forfait complet (100% pour toutes les étapes étudiées).

3.1.2 Deuxième type de forfait : la logique d'externalisation différenciée par étape

Les projets du deuxième groupe au forfait (forfait de réalisation) montrent une répartition différente : sur 10 projets, 6 sont des nouveaux développements. On retrouve également une tierce maintenance (délégation de la réalisation seulement, l'analyse des besoins restant en interne car l'entreprise souhaite garder la maîtrise des relations avec les utilisateurs) et 3 projets progiciels.

Dans ce groupe, c'est une logique de **gestion contractuelle différenciée selon les étapes** du processus de développement qui est appliquée (analyse du besoin en interne, forfait dans les étapes suivantes et surtout en réalisation).

3.2 L'interprétation des deux groupes de projets utilisant la régie (groupes 3 et 5)

Pour interpréter ces deux groupes, nous utiliserons le premier et le troisième plan principal (figure 8.14), qui en permettent la meilleure visualisation, puis le détail des projets (tableau 8.7).

La caractéristique de "projet en régie" commune aux deux groupes est très visible sur le premier plan principal qui les situe dans la même zone du plan (carré supérieur droit). Le troisième plan principal permet d'affiner l'analyse et sépare clairement les deux groupes :

- le groupe 3 est situé *en bas* (réalisation interne faible, conception et réalisation en régie) et à droite du plan (régie). Ce sont les "**régies de conception et de réalisation**", dans lesquelles la régie domine dès la phase de conception.

- le groupe 5 est également situé à droite (régie) du plan mais *en haut* (réalisation interne plus forte, analyse du besoin interne plus faible que dans le groupe précédent, analyse du besoin en régie). Ce sont pour nous les "**régies d'appoint**", qui recouvrent des projets où la régie a été utilisée en continu sur l'ensemble du processus de développement mais où la maîtrise du projet n'a pas été déléguée.

Représentation de la typologie sur le troisième plan principal

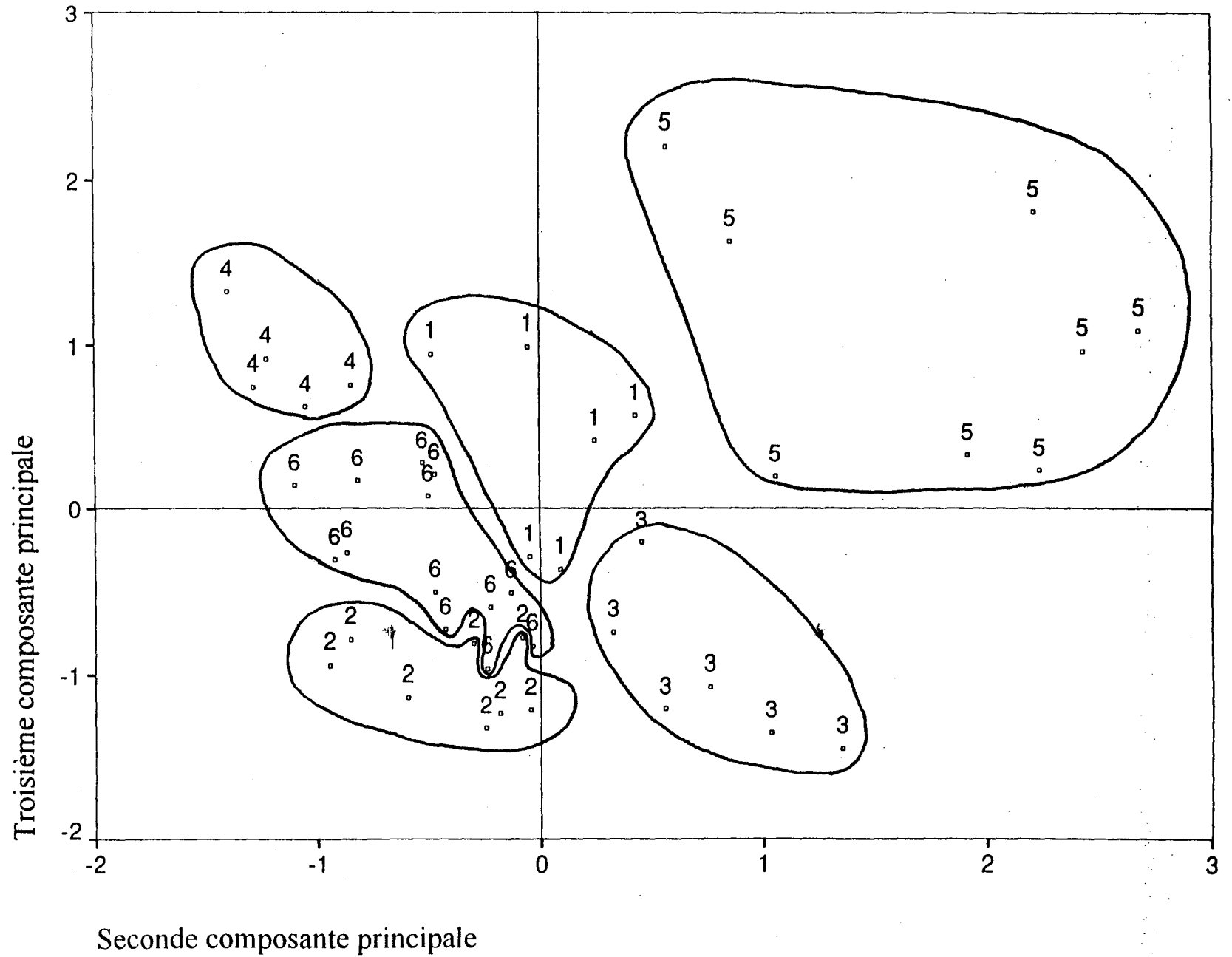


Figure 8. 14. Représentation de la typologie sur le troisième plan principal (axe 2; axe 3)

Le détail des projets concernés (cf. tableau 8.7.) permet de constater que, contrairement au cas des projets au forfait du groupe 1 (forfait de tout ou partie du projet), **l'analyse du besoin n'est jamais totalement déléguée**, même dans le groupe 5, et que l'informatique interne conserve une implication relativement forte en phase de conception, voire même dans certains cas en phase de réalisation.

groupe 3 - les "régies de conception et de réalisation"						
Projet	Analyse du besoin		Conception		Réalisation	
	% interne	% régie	% interne	% régie	% interne	% régie
3	100		50	50		100
6	100		30	70	10	90
22	100		50	50	20	80
23	100		50	50	20	80
28	100			100		100
33	100			100		100
34	100		20	80	50	50
37	100		50	50		100
60	100		20	80		100
groupe 5 - les "régies d'appoint sur tout le projet"						
Projet	Analyse du besoin		Conception		Réalisation	
	% interne	% régie	% interne	% régie	% interne	% régie
13	50	50	50	50	50	50
14	65	35	50	50	10	90
16	30	70	10	90		100
24	50	50	20	80		100
27	50	50	50	50	75	25
31	25	75		100		100
32	50	50		100		100
38	50	50	50	50	75	25
58	20	80	20	80	20	80

Tableau 8.7. Les deux groupes de projets en régie : détail des projets

Pour le groupe 3 (régie de conception et de réalisation), sur les 9 projets,

- 5 sont des nouveaux développements,
- 3 sont des maintenances évolutives,
- 1 projet correspond à des travaux complémentaires autour d'un progiciel.

Pour le groupe 5 (régie d'appoint), sur les 9 projets :

- 5 sont des nouveaux développements (avec dans deux cas une très forte part d'interfaces),
- 2 sont des maintenances évolutives,
- 2 correspondent à la réalisation d'interfaces autour d'un progiciel.

On voit que les types de projets concernés sont très proches dans les deux groupes.

Au-delà du détail des projets, la partition de la régie entre ces deux groupes correspond selon nous à deux approches de ce mode contractuel : dans un premier cas (groupe 3), on considère la régie comme une ressource spécialisée sur certaines étapes, plus techniques et surtout moins relationnelles ; dans le second cas (groupe 5), même si l'utilisation de la régie tend à croître dans les phases techniques, la différenciation entre ressource interne et régie est faible et les extérieurs ont accès au client interne (l'utilisateur). Ce dernier groupe semble donc montrer la possibilité d'une véritable interchangeabilité des deux modes d'organisation.

3.3 L'interprétation des deux groupes de projets "internes" (groupes 4 et 6)

Ces deux groupes de projets occupent sur le premier plan principal le carré inférieur droit (analyse du besoin, conception et réalisation interne). Le groupe 4 se détache sur le second plan principal du groupe 6 : il est situé "au-dessus", ce qui montre une plus grande utilisation du mode interne en phase de réalisation. Sur le troisième plan principal, les deux groupes sont également séparés, situés sur la gauche du plan (peu de régie), en haut pour le groupe 4 (réalisation et conception interne), proche de l'axe horizontal pour le groupe 6.

Nous interprétons ces deux groupes de la façon suivante :

- le groupe 4 est composé de projets étudiés, conçus et réalisés en interne. Ce sont les "tout interne", où l'informatique assume l'ensemble des tâches du processus de développement. L'externalisation est très marginale dans ce groupe.

- le groupe 6 est caractérisé par un recours à l'externalisation qui reste très faible non seulement dans la phase analyse du besoin, mais également en phase de conception. Ce n'est qu'en phase de réalisation que l'appel à l'extérieur devient prioritaire.

groupe 4 - les "tout interne"									
Projet	Analyse du besoin			Conception			Réalisation		
	% interne	régie	forfait	% interne	régie	forfait	% interne	régie	forfait
4	100			100			100		
10	100			100			100		
12	100			100			100		
18	100			85		15	85		15
39	100			100			75		25
49	100			100			100		
50	100			100			100		
51	100			100			70	30	
53	100			80	20		80	20	
groupe 6 - les "dominance interne avec externalisation limitée"									
Projet	Analyse du besoin			Conception			Réalisation		
	% interne	régie	forfait	% interne	régie	forfait	% interne	régie	forfait
5	100			90	10		30		70
7	100			100			30	30	40
15	100			80	20		10	90	
17	100			100			10	80	10
29	100			100			20	80	
36	100			100				100	
41	100			75	25		25	75	
42	100			75	25		25	75	
43	100			90		10	50		50
44	100			80	20		50	50	
45	100			100			50	50	
54	100			80	20		20	70	10
55	90		10	60		40	50	50	
56	90	10		100			40	60	

Tableau 8.8. Les deux groupes de projets gérés en interne

3.4 Synthèse de la typologie

Les 6 groupes identifiés se répartissent donc de la façon suivante :

	Groupe 1 Forfait global	Groupe 2 Forfaits de conception /réalisation	Groupe 3 Régies de conception /réalisation	Groupe 5 Régies d'appoint	Groupe 4 Tout interne	Groupe 6 Dominance interne
Mode dominant						
- analyse du besoin	forfait	interne	interne	mixte	interne	interne
- conception	forfait	mixte	régie	mixte/régie	interne	interne
- réalisation	forfait	forfait	régie	régie	interne	mixte
Rôle de l'informatique interne :						
- sélection fournisseur	oui (avec utilisateur)	oui	oui	oui	N/A	oui
- relations utilisateurs	faibles	oui	oui	oui	oui	oui
- responsabilité projet	non	oui	oui	oui	oui	oui
- implication technique	non	faible	limitée	limitée	forte	limitée
Type d'évaluation	résultat	résultat	comportement	comportement	comportement	comportement
Transfert de la responsabilité chez le fournisseur	oui (tout le projet)	oui (l'activité concernée)	non	non	non	non
Contacts entre le fournisseur et l'utilisateur interne	oui	non	non	oui	non	non
Différenciation du mode d'organisation selon l'étape du processus	oui (faible)	oui (fort)	oui (fort)	oui (faible)	non	oui (fort)

Tableau 8.9. Synthèse de la typologie

Nous terminons avec cette typologie l'analyse descriptive des choix d'externalisation. Nous allons maintenant tenter, dans la troisième section de ce chapitre, d'expliquer ces choix et de les lier aux différentes variables explicatives élaborées à partir de nos hypothèses.

Section 3 - Test des hypothèses

Nous allons maintenant chercher à lier les variables explicatives issues de notre jeu d'hypothèses avec les choix d'externalisation. Notre objectif est de déterminer si les résultats obtenus corroborent ou non les hypothèses émises, ce dans le cadre des projets étudiés. Rappelons que nous cherchons à expliquer les comportements rencontrés au sein de notre échantillon, sachant que des tests supplémentaires seraient nécessaires pour confirmer la validité externe des résultats obtenus. A cet égard, nous ne pouvons pour des raisons pratiques et théoriques que limiter l'ambition de ce travail à la définition d'un premier cadre d'analyse et à la réalisation de premiers tests de validation faible.

Nous cherchons à lier un **comportement en matière de choix d'externalisation** avec une série de variables potentiellement explicatives. Précisons qu'un seul jeu de réponses a été établi pour chaque projet. En particulier, concernant les raisons du choix, il eût été impraticable de demander à nos interlocuteurs de justifier séparément leurs décisions pour chaque étape du projet (analyse du besoin, conception, réalisation). C'est donc le **profil global d'externalisation** sur l'ensemble des étapes étudiées qui a été commenté.

Nous avons retenu pour cela une double approche :

- dans un premier temps, nous procédons à une étude des corrélations entre les coordonnées de nos axes factoriels, choisies comme variables dépendantes, et les variables explicatives. Les coordonnées d'un projet sur les axes sont une approximation acceptable du profil global d'externalisation. Nous cherchons à expliquer cette position sur les axes par les différents facteurs.

- dans un second temps, nous étudions le lien entre la typologie en six groupes précédemment décrite (variable qualitative) et nos variables explicatives. Nous avons pour cela procédé à une analyse de la variance.

Mais auparavant, nous allons analyser rapidement nos variables explicatives et en particulier étudier si certaines d'entre elles sont corrélées.

I. Analyse des variables explicatives ; étude des corrélations

1.1 Corrélations entre les variables descriptives du projet

	Contrainte budgétaire	Urgence	Nouveauté technologie	Importance stratégique	Liens avec le SI	Spécificité du domaine	Structure
Urgence	.20 (.12)						
Nouveauté technologie	.45 (.00)	.08 (.54)					
Importance stratégique	.14 (.28)	.13 (.31)	.13 (.34)				
Liens avec le SI	.14 (.29)	-.12 (.35)	-.07 (.60)	-.12 (.37)			
Spécificité du domaine	-.03 (.81)	-.17 (.19)	.39 (.00)	-.06 (.63)	-.01 (.94)		
Structure	.21 (.11)	.17 (.19)	.16 (.23)	-.00 (.99)	-.17 (.20)	-.02 (.85)	
Taille	.42 (.00)	.12 (.36)	.44 (.00)	.27 (.03)	-.00 (.99)	.10 (.44)	.25 (.06)

Les variables sont faiblement corrélées entre elles, avec quelques exceptions :

- la contrainte budgétaire sur le projet est plus forte pour des projets de grande taille ou présentant une grande nouveauté technologique. On peut penser que cette contrainte répond à une appréhension d'un risque plus grand sur le projet, risque de taille ou risque technique.

- une forte nouveauté technologique est corrélée positivement à la taille du projet et à la spécificité du domaine d'application. Cette deuxième corrélation est étonnante. On peut éventuellement l'expliquer par le fait que la technologie ouvre des nouveaux domaines d'activité, mais peut-être y a-t-il eu une partielle confusion des deux notions dans l'esprit de nos interlocuteurs.

- l'importance stratégique est liée à la taille.

- la taille est la variable la plus corrélée aux autres, ce qui explique sans doute sa présence dans certains modèles explicatifs.

1.2 Corrélations entre les variables d'évaluation du poids des facteurs dans la décision

	Coût production	Contraintes de délai	Optimisat. charge/capa	Spécificité technique	Impact stratégique	Connaiss. du SI	Meilleur contrôle
Contraintes de délai	- .16 (.23)						
Optimisat. charge/capa	- .25 (.06)	.31 (.02)					
Spécificité technique	.02 (.89)	.05 (.73)	- .12 (.36)				
Impact stratégique	.17 (.19)	.14 (.30)	- .14 (.28)	.10 (.43)			
Connaiss. du SI	.20 (.13)	- .20 (.13)	.09 (.49)	- .07 (.61)	- .16 (.24)		
Meilleur contrôle	.33 (.01)	- .28 (.03)	- .39 (.00)	- .17 (.18)	- .02 (.89)	- .01 (.97)	
Habitude	.07 (.61)	- .29 (.02)	- .10 (.46)	- .27 (.03)	.14 (.30)	.17 (.21)	.23 (.07)

Quelques variables sont corrélées entre elles :

- les critères de coût de production et de maximisation du contrôle sont convergents.
- un poids important accordé aux contraintes de délai est comme nous l'attendions lié au critère d'optimisation charge/capacité. Il est intéressant de voir que, dans ce cas, le poids du facteur contrôle est significativement plus faible (on note de même une corrélation négative entre ce dernier critère et l'optimisation charge / capacité). De même il est accordé moins de poids au respect des habitudes sous contrainte de délai.
- il est accordé moins de poids au respect des habitudes d'externalisation quand la spécificité technique est déterminante pour le choix.

Ce deuxième jeu de corrélations est intéressant car il nous permet de dégager certaines associations et oppositions dans les choix : les critères d'optimisation charge / capacité et de rapidité semblent s'exercer aux dépens de la recherche du meilleur contrôle. Le respect des habitudes est moins important en cas d'urgence ou de spécificité technique.

1.3 Corrélations entre les deux groupes de variables

	Coût product.	Cont. de délai	Optimis. ch./capa	Spécificité technique	Impact strat.	Connaiss. du SI	Meilleur contrôle	Habitude
Contrainte budgétaire	.25 (.05)	-.00 (.98)	-.29 (.03)	.15 (.24)	.03 (.80)	.05 (.70)	.18 (.16)	-.10 (.46)
Urgence	.13 (.32)	.33 (.01)	-.08 (.52)	-.18 (.18)	.33 (.01)	-.08 (.53)	.04 (.77)	.05 (.68)
Nouveauté technologie	.16 (.21)	-.04 (.75)	-.40 (.00)	.43 (.00)	.11 (.39)	-.02 (.86)	.11 (.40)	-.17 (.20)
Importance stratégique	.16 (.22)	.24 (.07)	-.03 (.82)	-.05 (.73)	.37 (.00)	-.20 (.13)	.17 (.19)	-.14 (.27)
Liens avec le SI	.24 (.07)	-.11 (.39)	.10 (.47)	-.12 (.36)	-.16 (.22)	.49 (.00)	.08 (.55)	-.01 (.92)
Spécificité du domaine	-.22 (.09)	-.26 (.05)	-.19 (.15)	.12 (.38)	-.13 (.31)	.11 (.42)	.13 (.32)	-.04 (.75)
Structure	.26 (.05)	.13 (.32)	-.10 (.45)	.03 (.85)	.17 (.19)	-.10 (.46)	.17 (.21)	.09 (.50)
Taille	.25 (.05)	-.03 (.80)	-.31 (.01)	.24 (.07)	.36 (.01)	-.22 (.10)	.28 (.03)	-.04 (.77)

La première constatation est que les **variables illustrant un même concept sont corrélées positivement**, ce qui est cohérent et correspond à notre attente : spécificité technique et nouveauté technologique, importance du critère stratégique et importance stratégique du projet, importance de la connaissance du système d'information actuel et liens du projet avec le système d'information, contrainte de délai et urgence.

L'analyse de ces corrélations est de plus intéressante car elle nous donne quelques indications sur les liens entre le poids accordé aux différents critères de choix d'externalisation et les caractéristiques du projet.

- le poids accordé au coût de production est plus important pour des projets bien structurés et de grande taille.

- le poids accordé aux contraintes de délai est moins important quand la spécificité du domaine couvert par le projet est élevée.

- le poids accordé à l'optimisation charge / capacité est plus important pour les projets de petite taille, soumis à de faibles contraintes budgétaires et présentant une faible nouveauté technologique.

- la recherche d'un meilleur contrôle concerne davantage les projets de grande taille.
- le poids accordé aux considérations stratégiques est plus important pour les projets urgents et de grande taille.

L'analyse des corrélations entre variables explicatives est globalement conforme à notre attente: les mêmes concepts sont illustrés par des variables cohérentes entre elles et le reste des variables sont globalement peu corrélées entre elles. Au sein du groupe des variables descriptives des projets, les variables taille et nouveauté technologique sont les plus corrélées aux autres. Au sein du groupe des variables d'évaluation du poids des facteurs, les variables d'importance des contraintes de délai et de maximisation du contrôle sont les plus corrélées aux autres. Entre les deux groupes de variables, c'est l'importance des facteurs stratégiques et celle de l'ajustement charge / capacité qui sont les plus liées aux caractéristiques des projets.

II. Etude des corrélations entre les variables explicatives et les deux premiers axes principaux

Nous allons maintenant voir s'il est possible de lier globalement le recours aux trois modes d'organisation à nos variables. Pour cela, nous allons chercher à expliquer les premier et le second axe de notre analyse en composantes principales. Nous avons en effet vu que l'axe 1 oppose le forfait (côté négatif de l'axe) aux autres modes (côté positif de l'axe). L'axe 2 oppose quant à lui la régie (côté positif de l'axe) à l'informatique interne (côté négatif de l'axe).

2.1 Corrélations avec le premier axe principal (forfait/autres modes)

Nous avons repris dans le tableau 8.10 les corrélations significatives entre les variables des trois groupes et le premier plan principal. Une corrélation négative significative est liée au choix du forfait. Une corrélation positive significative est liée au choix de l'informatique interne ou de la régie.

	forfait (<0)	interne, régie (>0)
1. Variable		
Nouveauté technologique	-33 (.01)	
Spécificité technique	-26 (.04)	
Structure du projet	-35 (.01)	
Spécificité du domaine	-28 (.03)	
Solution habituelle		.29 (.03)

Tableau. 8.10. Corrélations entre le premier axe principal et les variables explicatives

Les résultats obtenus corroborent certaines hypothèses de la recherche.

Le choix du forfait est fortement lié au **caractère technologique** du projet (H2). Les entreprises font appel à une externalisation de spécificité qui leur permet de profiter des progrès technologiques tout en évitant d'investir dans la recherche ou la formation de profils d'informaticiens trop spécialisés.

L'externalisation au forfait est fortement liée au **caractère structuré** du projet (H6). Il semble ainsi que quand le projet est très structuré, les entreprises choisissent de façon prioritaire un mode de pilotage au résultat.

Les critères d'importance stratégique, d'ajustement charge/capacité, de délais, d'intégration au système d'information ne sont en revanche pas significativement corrélés au choix du forfait.

Notons également que la spécificité du domaine apparaît liée au choix du forfait, mais dans un sens contraire à notre hypothèse (H7). Une forte spécificité est liée à l'utilisation du mode forfaitaire : nous ne savons comment interpréter ce résultat qui est contraire à ce que suggère l'analyse des coûts de transaction (la forte spécificité des actifs est théoriquement favorable au choix d'une structure de gouvernance interne). Il faut sans doute remettre en cause ici la qualité de la variable pour illustrer le concept, ainsi que la confusion possible avec le caractère de spécificité technologique ou technique, avant de tirer des conclusions de ce résultat.

Enfin, cette analyse des corrélations souligne la **rupture avec les habitudes** que représente le mode forfaitaire. Ce critère est en effet le seul qui a une corrélation significativement contraire à la mise en oeuvre de forfaits.

2.2 Corrélations avec le second axe principal (régie/interne)

Le tableau 8.11 reprend les corrélations avec le second axe principal, qui distingue les modes régie (côté positif de l'axe) et interne (côté négatif).

	interne (<0)	régie (>0)
1. Variables		
Optimisation charge / capacité		.32 (.01)
Importance du critère de coût	-.45 (.00)	
Contraintes de délai		.43 (.00)
Liens avec le SI existant	-.31 (.02)	
Contrôle du projet	-.30 (.02)	

Tableau. 8.11. Corrélations entre le second axe principal et les variables explicatives

Le choix de la **régie** est corrélé à l'**ajustement charge / capacité** (H3) et aux contraintes de **délais** (H4). Ceci souligne l'opposition entre les deux modes d'externalisation étudiés : le forfait semble répondre à un problème de technologie et être utilisé dans le cadre de projets bien structurés, la régie répond aux problèmes d'ajustement de charge.

Le choix de l'informatique **interne** est corrélé à trois critères :

- l'importance de l'**intégration du projet au système d'information existant** (H5). On délègue moins à l'extérieur ce qui est très intégré,

- importance du critère de **coût** : il semble que la comparaison coût interne / coût de la régie soit à l'avantage des équipes internes ; notons que cette évaluation a été établie sur des bases "intuitives" ("je pense que c'est moins cher en interne") plutôt que sur une comparaison chiffrée et complète des différents éléments,

- enfin, il y a une corrélation entre la recherche d'un **meilleur contrôle du projet** et le choix de l'interne plutôt que de la régie. On est ici dans le cas d'un choix entre d'une part une organisation hiérarchique, fonctionnant sur le principe de l'adaptation coopérative et d'un contrôle continu du comportement et d'autre part un mode d'organisation hybride (régie), fonctionnant sur un système de pilotage mixte alliant quelques contraintes de marché à un système de contrôle administratif. On peut penser que le bénéfice des contraintes de marché exercé sur la régie ne parvient pas à compenser les avantages de l'adaptation interne. De plus, la tension actuelle sur les coûts fait que le recours à la régie est moins systématiquement récurrent, ce qui tend à réduire les capacités d'adaptation coopérative entre le fournisseur et son client⁸.

2.3 Synthèse de l'étude des corrélations

Le tableau 8.12 présente une synthèse des résultats.

Hypothèse	Résultat	Conformité à l'hypothèse ?
H1 Importance stratégique	non vérifié	non
H2 Nouveauté technologique	corrélé positivement à une externalisation au FORFAIT	oui
H3 Ajustement charge/capacité	corrélé positivement à une externalisation en REGIE	oui
H4 Contraintes de délais	corrélé positivement à une externalisation en REGIE	oui
H5 Intégration au système d'information	corrélé positivement au mode INTERNE	oui
H6 Degré de structure du projet	corrélé positivement à une externalisation au FORFAIT	oui
H7 Spécificité du domaine d'application	corrélé positivement à une externalisation au forfait	non

Tableau 8.12. Synthèse des résultats de l'étude des corrélations

⁸ . Nous reviendrons sur ce point dans l'étude qualitative (chapitre 9).

Les hypothèses sont globalement vérifiées, à l'exclusion de deux d'entre elles :

- le critère d'importance stratégique des projets ne s'est pas avéré discriminant,
- le critère de spécificité du domaine d'application donne des résultats contraires à notre attente ; davantage que l'hypothèse, nous pensons que c'est l'indicateur choisi qui doit être remis en question (formulation peu claire).

L'élément le plus intéressant nous semble être la différence des critères intervenant respectivement dans le choix d'une externalisation en régie et au forfait :

- concernant la régie, on trouve des critères opérationnels relevant de l'approche industrielle (délais, ajustement de charge),
- concernant le forfait, on trouve un critère lié à l'approche industrielle (technologie), mais aussi un critère lié à une problématique de contrôle (degré de structure du projet).

Enfin, le critère d'intégration du projet au système d'information existant (critère issu à la fois de l'approche industrielle et des théories du contrôle) est corrélé, comme nous l'attendions, avec le choix de l'internalisation.

III. Le lien entre les variables explicatives et la typologie : analyse de la variance

Nous avons réalisé une analyse de la variance entre les variables explicatives de notre modèle et la typologie des projets en six groupes. L'objectif est de déterminer d'une part si la différence entre les groupes est globalement significative et d'autre part s'il existe des groupes qui, comparés deux à deux, sont significativement différents (le test utilisé est le test Tuckey-HSD avec un niveau de signification de 0.05).

Il s'agit donc ici non seulement de différencier les trois modes (interne, régie, forfait) comme dans l'analyse précédente, mais d'affiner cette analyse pour tenter d'expliquer les différences entre les six groupes identifiés.

3.1 Résultats de l'analyse de variance

Les résultats de l'analyse de variance font apparaître des différences significatives sur les facteurs suivants :

Variables	F ratio	Niveau de signification	Identification de groupes différents
Nouveauté technologique	3.19	.0137 **	groupes 1 et 3 **
Optimisation charge / capacité	3.51	.0082 **	groupes 3 et 6 **
Contraintes de délais	2.43	.0469 **	non
Solution choisie pour minimiser le coût	3.79	.0052 **	groupes 3 et 4 **
Solution habituelle	2.09	.0807 *	non
* faiblement significatif (<0.10) ; ** très significatif (<0.05)			

Tableau 8.13. Résultats de l'analyse de variance

Le tableau 8.13 indique les valeurs moyennes des variables pour les différents groupes et met en évidence les différences significatives pour les groupes pris deux à deux :

	Valeurs moyennes pour les différents groupes					
	Groupe 1 Forfait global	Groupe 2 Forfait conc/réa	Groupe 3 Régies conc/réa	Groupe 4 Tout interne	Groupe 5 Régies d'appoint	Groupe 6 Dominance interne
Technologie	3.89 **	3.70	1.78 **	2.67	2.78	2.21
Ajust. charge / capacité	2.00	2.10	3.00 **	2.33	2.78	1.86 **
Contraintes de délais	1.67	1.90	2.33	1.33	2.22	1.64
Coût	1.44	1.80	1.11 **	2.22 **	1.33	1.93
Solution habituelle	1.22	1.50	1.56	2.00	1.67	2.14

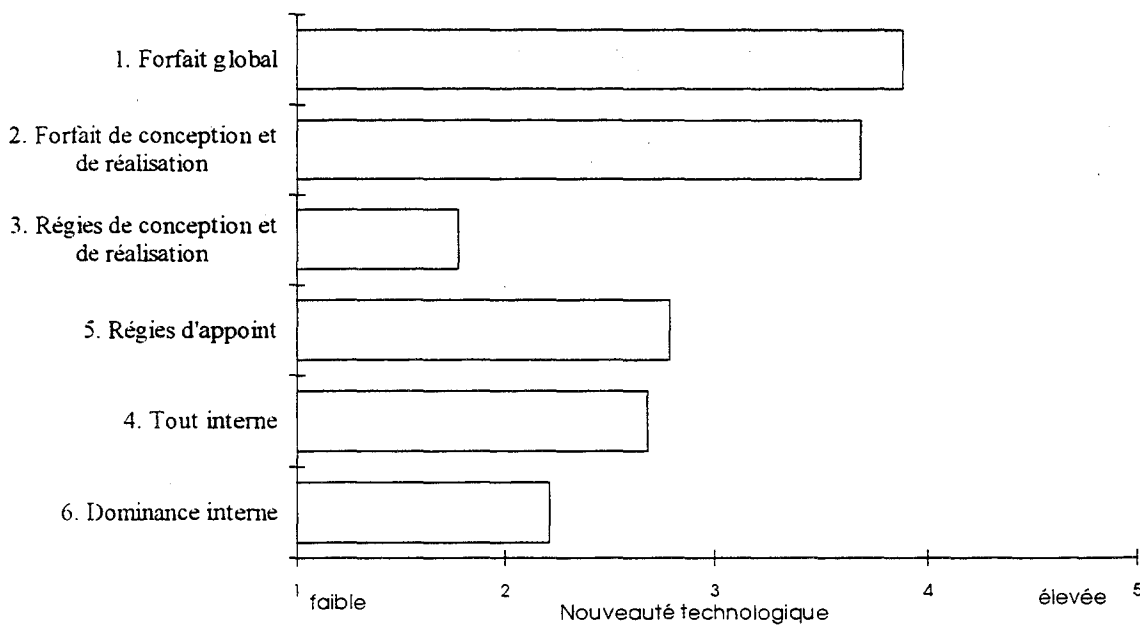
Tableau 8.14. Valeurs moyennes des variables explicatives pour les différents groupes

Les résultats de l'analyse de variance ne font pas apparaître de différence significative entre les six groupes de projets sur les critères d'importance stratégique, d'intégration au système d'information existant, de structure et de spécificité du domaine d'application.

2.2 Interprétation des résultats

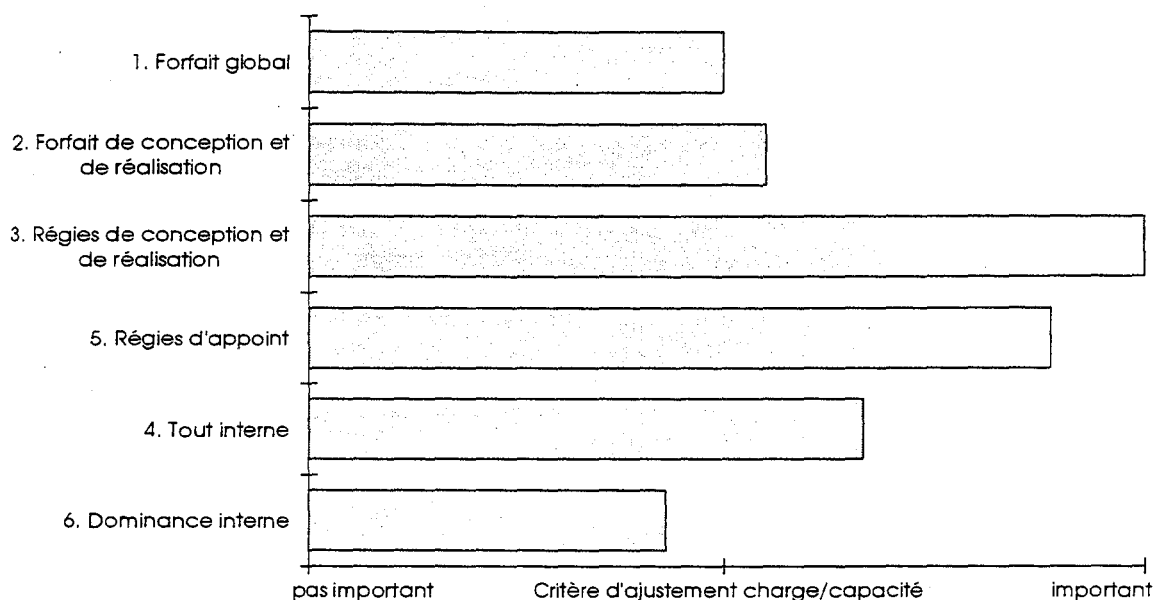
Trois hypothèses donnent des résultats globalement significatifs, concernant l'impact de la technologie, de l'ajustement charge/capacité et des contraintes de délais.

La nouveauté technologique du projet est variable selon les groupes. En particulier, les deux groupes au forfait sont composés de projets à forte nouveauté technologique. Notre interprétation est que les entreprises se sentent sans doute moins à même de maîtriser un projet faisant appel à une technologie nouvelle pour elles ; quand elles choisissent l'externalisation, elles cherchent alors à pallier l'incertitude technologique en demandant à leur fournisseur de s'engager sur un résultat (forfait). Un autre résultat intéressant apparaît ici, à savoir le lien entre une faible nouveauté technologique et une externalisation de type régie de conception et de réalisation.

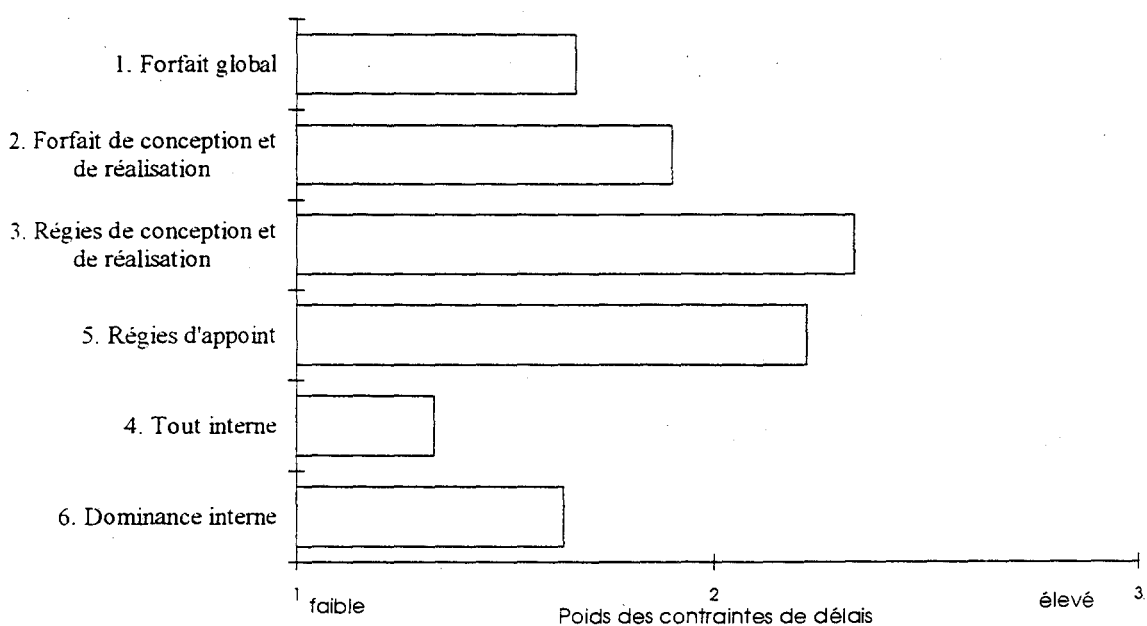


L'optimisation de la capacité est mise en avant dans le cadre de régies de conception et de réalisation (groupe 3). Elle est en revanche considérée comme un critère peu important dans le cas de projets à dominance interne avec appel limité à l'externalisation (groupe 6). Le recours massif à la régie dans les phases de conception et de réalisation est justifié par un

problème de capacité interne ; l'utilisation de la régie est sélective sur ces deux phases, puisque l'analyse du besoin n'est pas concernée. On retrouve ici un problème bien connu de l'activité de développement qui est la très forte variation du niveau d'activité selon le nombre et la taille des projets retenus, variation surtout sensible en phase de conception et de réalisation.

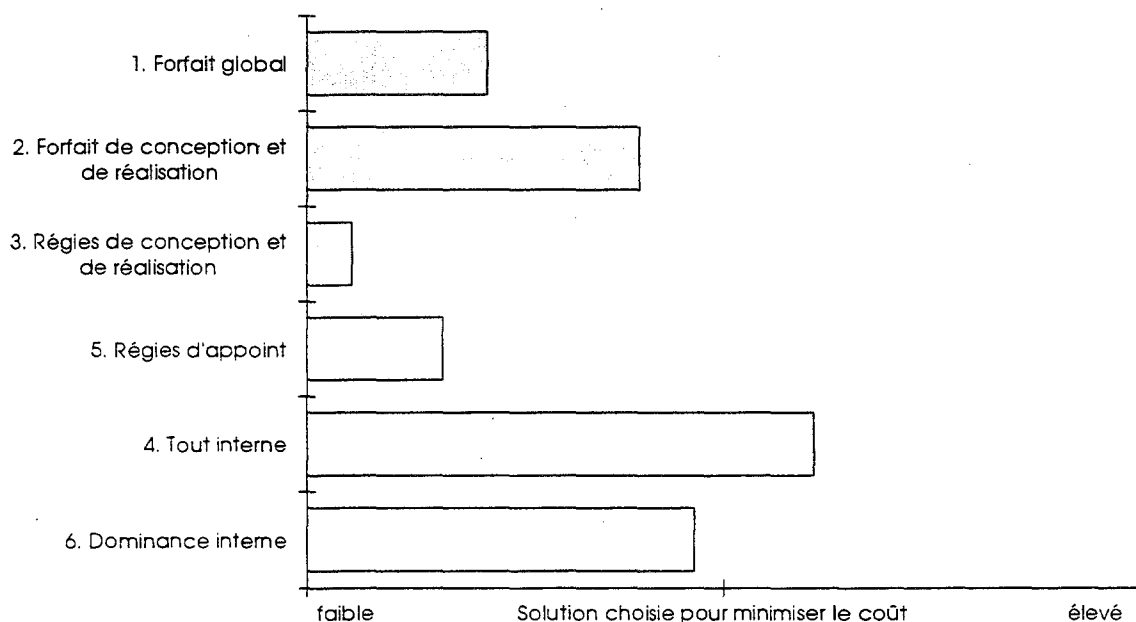


L'importance des contraintes de délais va dans le même sens que le critère précédent ; elle concerne principalement l'externalisation en régie.

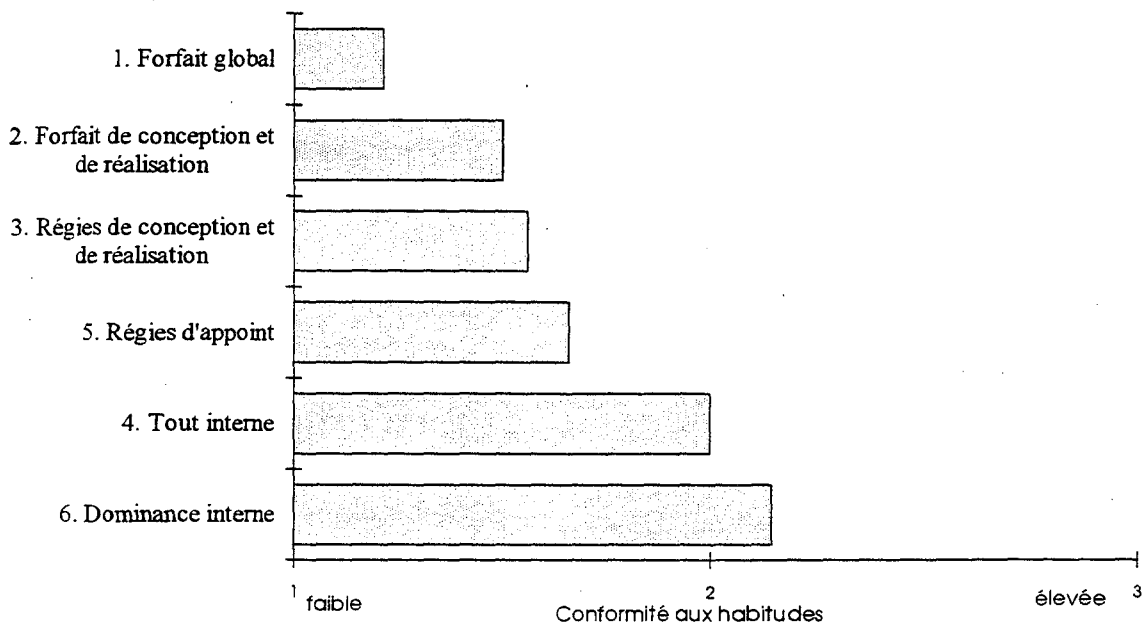


Deux variables complémentaires donnent également des résultats intéressants.

Le critère de **coût de production** est jugé très important dans le cas de recours quasi exclusif à l'informatique interne. Il obtient en revanche un très faible score en cas de régie de conception et de réalisation. L'informatique interne est perçue comme étant moins chère que la régie. Nous avons pensé qu'une meilleure spécialisation des fournisseurs pouvait conduire à des réductions de coût. L'opinion exprimée par les personnes ayant répondu au questionnaire semble être au contraire que l'utilisation de l'informatique interne répond à un souci de réduction du coût, alors que ce critère n'est pas mis en évidence en cas de recours à l'externalisation. On peut penser que les informaticiens internes ont ainsi cherché à valoriser le recours à leurs propres services. Il est également très intéressant de constater que le critère de coût n'est pas mis en avant dans le cas de recours à l'externalisation au forfait.



Enfin, la diversité des réponses sur le critère d'habitude (question : "c'est la solution habituelle pour nos développements") est également intéressante : on voit que les solutions internes sont perçues comme "habituelles" ; la nouveauté du forfait dans les pratiques apparaît également très clairement.



L'analyse de la variance met en évidence des facteurs intéressants ; nous considérons les résultats obtenus comme encourageants et susceptibles de fournir une première base pour des recherches futures :

- lien entre la nouveauté technologique et les choix d'externalisation au forfait,
- poids des facteurs opérationnels (capacité, délai) sur les choix d'externalisation en régie,
- perception de l'informatique interne comme une solution peu coûteuse, notamment en comparaison avec la régie,
- rupture des habitudes avec le mode forfaitaire.

Conclusion

Le premier objectif de l'étude quantitative était de décrire les pratiques existantes ; nous avons donc utilisé des méthodes statistiques descriptives pour représenter les choix d'externalisation menés sur 60 projets qui ont composé notre échantillon.

Les résultats obtenus ne sont pas directement généralisables ; nous pensons néanmoins qu'ils représentent un premier pas dans l'étude de ce problème et peuvent constituer des pistes de recherche pour les travaux suivants.

Concernant la forme prise par l'externalisation, nous avons fait les observations suivantes :

- l'externalisation est **variable selon les étapes du développement** étudiées : l'appel à l'informatique interne domine dans la phase d'analyse du besoin, les solutions mixtes "interne/régie" et "interne/forfait" dominent en phase de conception et de réalisation ; dans cette étape cependant, on constate également des solutions d'externalisation complète au forfait ou en régie.

- le choix du **mode de pilotage** du fournisseur extérieur est **généralement conservé au long des trois étapes étudiées** : un fournisseur piloté en régie en phase de conception reste piloté en régie en phase de réalisation (et de même pour le forfait).

Nous avons dégagé une **typologie des projets en six classes**, selon l'importance et le mode d'externalisation choisis :

- le **"forfait global"** : toutes les étapes du projet, y compris l'analyse du besoin sont externalisées avec un mode de pilotage du fournisseur au résultat,

- le **"forfait de conception et de réalisation"** : l'analyse du besoin est faite en interne, une équipe mixte forfait/interne est utilisée en phase de conception, la réalisation est externalisée au forfait - on a une "montée en puissance de l'externalisation" dans les trois premières étapes,

- la **"régie de conception et de réalisation"** : l'analyse du besoin est faite en interne, ce sont les équipes en régie qui font les deux étapes suivantes - la régie est utilisée pour les étapes techniques,

- la "**régie d'appoint**" : l'apport de la régie se fait tout au long du projet, il semble qu'il y ait une relative interchangeabilité des modes interne et régie,
- le "**tout interne**" : toutes les phases du développement sont assurées en interne,
- la "**dominance interne**" : la plus grande partie du développement est assurée en interne, malgré une montée de l'externalisation en phase de réalisation.

Nous avons ensuite cherché à voir dans quelle mesure certaines variables peuvent, sur notre échantillon, expliquer les choix d'externalisation.

Nous avons tout d'abord étudié le lien entre les variables et les deux premiers axes de notre analyse en composantes principales :

- L'axe 1 oppose le **forfait** aux autres modes ; le côté négatif de l'axe (choix du forfait) est corrélé avec deux variables de notre modèle, la **technologie** et la **structure** du projet ; il est opposé au critère "de respect des habitudes" dans la décision.

- L'axe 2 oppose la **régie** à l'**informatique interne** ; du côté de la régie, on trouve le critère d'**optimisation charge / capacité** et les contraintes de **déla**i. Du côté de l'informatique interne, on trouve les critères d'optimisation du coût, de **liens du projet avec le système d'information** existant et de contrôle du projet.

Nous avons ensuite cherché à lier la typologie en six classes avec nos variables :

- Le degré de **nouveauté technologique** apparaît significativement différent selon les groupes. Il est en particulier plus élevé pour les groupes au forfait ; deux hypothèses peuvent être émises : d'une part la possibilité d'évaluer le fournisseur au résultat (forfait) peut effectivement être meilleure quand le projet est technique (fournisseur très spécialisé, maîtrisant bien la technologie), d'autre part le choix du forfait peut être une volonté de report du risque technologique sur le fournisseur.

- Le critère d'**optimisation charge/capacité** est plus élevé pour la "régie de conception/réalisation" : on retrouve ici l'utilisation classique de la régie comme moyen d'absorber les pointes de charge.

- Le critère d'optimisation du coût a été mis en avant dans le cas du "tout interne" : il semble que l'externalisation soit perçue comme plus chère que l'appel à la ressource interne.

Au total, l'externalisation en régie et au forfait semblent répondre à des logiques différentes:

- concernant la régie, les résultats font apparaître le poids de facteurs **opérationnels** (capacité, délais),

- concernant le forfait, on constate un lien avec la **nouveauté technologique** des projets (facteur industriel) et également avec le degré de **structure** du projet (facteur de contrôle).

A noter que le critère d'importance stratégique du projet n'apparaît pas au travers de nos résultats comme un facteur de différenciation des choix d'externalisation.

Nous allons maintenant chercher à compléter ces résultats quantitatifs et à en approfondir l'interprétation par l'analyse du discours sur l'externalisation : c'est l'objet de la synthèse des entretiens présentée dans le chapitre 9.

ANNEXE : Questionnaire sur les projets

① **NOM et Description rapide application**
(domaine, objectif)

En service : oui / non

- initiateur du projet (informatique / utilisateurs)

- taille - ordre de grandeur, en : M/H, ou : J/H, ou Budget :

② **TYPE DE PROJET :** 1. Maintenance évolutive, 2. Développement spécifique nouveau, 3. Interfaces, 4. Développements complémentaires ou interfaces autour d'un progiciel

③ **CARACTERISTIQUES de l'application**

	faible		moyen		fort
	--	-	=	+	++
- taille relative					
- urgence					
- importance stratégique					
- spécificité du domaine d'application					
- liens avec les applications existantes					
- nouveauté technologique					
- poids des contraintes budgétaires					
- projet peu structuré (--), très structuré (++)					

④ **RESSOURCES UTILISEES :**

Quels sont les moyens utilisés pour réaliser le projet
(en cas de plusieurs sources, indiquez le % du budget représenté par chaque source)

	analyse du besoin	conception	réalisation
Informatique interne	%	%	%
Appel à l'extérieur en régie	%	%	%
Appel à l'extérieur au forfait	%	%	%
	100%	100%	100%

⑤ **RAISONS DU CHOIX DES RESSOURCES UTILISEES (interne, régie, forfait)**

(indiquez si vous pensez que ces facteurs *ont influencé le choix entre développement interne, appel à l'extérieur en régie ou au forfait*)

	pas déterminant			facteur déterminant	
	--	-	=	+	++
-le choix des ressources a été dicté par l' urgence du projet					
-nous avons choisi les ressources pour minimiser le coût					
-l' importance stratégique de l'application a pesé sur le choix des ressources					
-le choix est dû à la capacité informatique interne (nb jours disponibles)					
-le choix est dû à la spécificité technique (outils, environnement, logiciels)					
-la connaissance du système d'information de l'entreprise a été déterminante					
-le choix de ces ressources permettait un meilleur contrôle du projet					
-c'est la solution habituelle pour nos développements					
Autres :					
-					
-					
-					

⑥ **RESULTATS ATTEINTS. Peut-on dire que :**

Plutôt Non A peu près Tout à fait

- les délais ont été respectés (sont respectés pour l'instant)			
- l'application satisfait totalement les utilisateurs			
- les coûts ont été respectés (sont respectés pour l'instant)			

Si c'était à refaire, vous choisiriez le même type de fournisseur (info interne, régie, forfait) : **oui / non**

Si non, préciser lequel :

Pourquoi?

⑦ **EN CAS DE RECOURS A UN FOURNISSEUR EXTERIEUR :**

Avez-vous fait un appel d'offres ?	oui	non
Aviez-vous déjà travaillé avec ce fournisseur ?	oui	non
Est-il un spécialiste de votre secteur d'activité ?	oui	non
Est-il un spécialiste de ce type d'application ?	oui	non
Utilise-t-il les mêmes outils (langages) que l'informatique interne ?	oui	non
Avez-vous modifié en cours de route le contrat de départ ?	oui	non
Si oui, sur quels points (% de modification)		
coûts : %; délais : %; contenu de l'application : %		

Si vous avez utilisé de la régie :

Vous êtes-vous mis d'accord avec le fournisseur sur un plafond	oui	non
Les dépassements sont-ils à la charge du fournisseur?	oui	non

Si vous avez utilisé un forfait :

Avez-vous adjoint un informaticien interne à l'équipe du fournisseur?	oui	non
Avez-vous contrôlé directement l'organisation des travaux du fournisseur?	oui	non
Le fournisseur a-t-il fait du prototypage?	oui	non

Chapitre 9 :
Résultats de la recherche qualitative

Introduction

L'analyse qualitative, dont la synthèse va être présentée dans ce chapitre, est le résultat de 20 entretiens, conduits dans les 12 entreprises auprès desquelles nous avons également collecté les données quantitatives. Ces entretiens ont duré entre une et quatre heures, ils ont tous été enregistrés sauf trois (dans un cas à cause d'un problème technique d'enregistrement, dans deux cas parce que nos interlocuteurs ont souhaité ne pas être enregistrés) ; on s'est alors reporté à la prise de notes effectuée systématiquement pendant tous les entretiens. Les entretiens que nous avons menés sont de type semi-directif. Nous avons utilisé un guide d'entretien dont un exemplaire est fourni en annexe. Ce guide nous a servi de rappel sur l'ensemble des thèmes à aborder pendant l'interview : nous avons en effet privilégié une forme peu directive, en laissant à nos interlocuteurs le choix de l'ordre des thèmes ; de plus, nous avons systématiquement utilisé des techniques de reformulation pour approfondir les aspects évoqués.

Notre objectif dans cette partie de la recherche terrain a été de dépasser le cadre du projet, analysé dans l'étude quantitative, pour nous consacrer à **l'étude du contexte dans lequel s'inscrivent les choix et mettre ainsi en évidence les facteurs environnementaux.**

La présentation des résultats obtenus est structurée en trois parties qui reprennent les trois questions qui ont dirigé notre recherche théorique.

1. Quels sont les **objectifs** assignés à l'informatique en général et à l'activité de développement d'applications en particulier ? Nous cherchons ici à déterminer dans quelle mesure cette activité ou une partie d'entre elle est considérée ou non comme **stratégique**, pour évaluer notre hypothèse théorique qu'une haute valeur stratégique est favorable à l'internalisation d'un processus.

2. Quelles sont les **contraintes industrielles** : comment nos interlocuteurs envisagent-ils l'évolution des volumes de production (pérennité de la demande) et quelle est leur attitude face à la technologie et ses évolutions ? Nous voulons ici tester les hypothèses concernant l'avantage industriel éventuel de fournisseurs extérieurs.

3. Comment est effectué le **contrôle** de l'activité ? Dans quelle mesure les choix d'externalisation sont-ils déterminés non seulement par des considérations stratégiques et industrielles, mais aussi par l'existence de coûts de pilotage (coûts de transaction) que le choix d'un mode de gouvernance particulier permet de minimiser ? Nous voulons ici déterminer dans quelle mesure les hypothèses liées aux modes de contrôle de l'activité sont ou ne sont pas vérifiées.

La dernière section de ce chapitre a pour titre "maîtriser l'externalisation", elle est consacrée au rôle de l'informatique interne dans la pratique de l'externalisation.

Les discours de nos interlocuteurs et les situations observées dans les différentes entreprises présentent beaucoup de similitudes et une certaine homogénéité des perceptions ; c'est pourquoi nous avons privilégié un exposé commun des résultats, structuré par thème (les objectifs, les contraintes industrielles, les modes de contrôle). A chaque fois qu'une situation nous a paru s'écarter de la référence commune, nous l'avons signalé dans l'exposé du thème concerné.

Ce chapitre donne une très large part aux citations des propos de nos interlocuteurs. En effet, il nous a semblé utile de leur "laisser la parole" afin de rendre compte de leurs opinions et de leurs perceptions de l'externalisation et de confronter leur vécu quotidien aux différents thèmes de l'approche théorique. Nous avons après chaque citation indiqué la source correspondante, en utilisant la codification présentée dans le tableau 8.1 : la lettre fait référence à l'entreprise concernée, le chiffre permet de situer le cas échéant différents interlocuteurs au sein de la même entreprise.

Section 1 - Le lien avec la stratégie : les objectifs assignés à l'activité de développement

Pour analyser les objectifs liés à l'activité de développement, nous avons demandé à nos interlocuteurs de préciser les missions assignées par la direction générale à leur fonction ; nous avons cherché à étayer les réponses obtenues par l'examen de documents internes, quand ils existaient ("charte informatique", plaquette de présentation des activités informatiques) et également par l'examen des plaquettes annuelles, en étudiant l'éventuelle section consacrée dans les commentaires aux activités informatiques. Nous avons également demandé à nos interlocuteurs de nous décrire le portefeuille d'applications de l'entreprise et de préciser l'éventuel rôle stratégique de celles-ci. Enfin, nous avons cherché à identifier les règles de gestion de l'activité : orientations, directives et consignes données par la direction générale.

L'exposé des résultats est structuré en trois parties. Dans un premier temps, nous nous attachons à décrire quelle est la perception de l'activité de développement dans les entreprises étudiées. Nous examinons ensuite successivement les deux objectifs principaux assignés à cette activité : développer la synergie avec le reste de l'entreprise, limiter les coûts.

I. La perception du rôle et de la place de l'activité de développement

1.1 Une mission de support non directement stratégique

Conformément à notre hypothèse concernant le choix des secteurs d'activité, tous nos interlocuteurs décrivent leur fonction comme jouant un rôle de **support** non directement stratégique. Pour argumenter leur position, ils se réfèrent principalement au métier, au secteur d'activité de leur entreprise, rejoignant en cela l'analyse de Porter :

"Dans notre métier, on ne pense pas que l'informatique est stratégique" (G)

"On est dans l'industrie, pas dans la banque ou l'assurance, donc on n'a pas une informatique stratégique ; on a une informatique de support pour des aspects informatiques de gestion" (H)

"Dans le management du groupe il n'y a pas encore une culture informatique, cela change avec l'arrivée des nouveaux dirigeants, petit à petit, mais l'informatique chez nous n'est pas aussi stratégique qu'elle peut l'être chez les banquiers, les assureurs, les transporteurs..." (B)

"Dans notre activité l'informatique n'est pas quelque chose d'essentiel. C'est vraiment un service au service des autres services. Mais on n'est pas moteur". (E₂)

Seule exception à ce concert quasi unanime, le discours de l'un de nos interlocuteurs, responsable des applications de son groupe : *"L'informatique peut tout à fait nous permettre de nous différencier par rapport à nos concurrents, tant sur le plan de l'efficacité [contrôle des coûts] que du développement commercial ; c'est une activité stratégique"*. Ce responsable - diplômé d'une grande école de gestion - s'est lancé depuis quatre ans dans une opération de "vente interne". Celle-ci semble porter ses fruits : *"quand on met sous les yeux des dirigeants tout ce qui a été fait, tout ce qui est en chantier, à l'occasion de réunions, ils comprennent très bien"* ; cependant la perception du rôle de l'informatique par les dirigeants reste celui d'une activité secondaire : *"Est-ce que la direction générale perçoit l'intérêt stratégique de l'informatique ? Je vais vous faire une réponse de normand : oui et non..." (K)*

Dans l'ensemble, nos observations montrent une activité informatique perçue comme un moyen, certes nécessaire, mais somme toute secondaire pour atteindre les objectifs des groupes étudiés, ce qu'exprime très clairement un responsable informatique :

"Nos facteurs clés de succès, c'est livrer des produits de qualité à un coût raisonnable dans des délais imposés. Alors l'informatique peut être un élément excellent pour notre position concurrentielle dans l'hypothèse où l'on a su rendre ces objectifs majeurs, dans la

mesure où l'on est capable de mettre en place de l'EDI¹ avec nos clients, nos fournisseurs, de mettre en place des systèmes de costing, de bien maîtriser le pilotage des machines... Elle fait partie d'un ensemble d'éléments concourant à la qualité et à la productivité, mais la qualité de l'informatique ne remplacera jamais la vision du manufacturing... C'est une fonction de support qui, quand elle joue bien son rôle, est complètement en phase avec le métier. C'est le moyen d'aller vers nos objectifs de manufacturing". (F)

1.2 L'incertitude sur la pérennité de l'activité de développement au sein de l'organisation

Les résultats qui sont présentés dans cette partie ont constitué pour nous une relative surprise ; l'honnêteté nous oblige à reconnaître que nous n'avions pas anticipé l'ampleur du phénomène exposé ici.

L'étude consacrée par l'OCDE (1985) aux logiciels décrivait un contexte de pénurie. La demande des entreprises en logiciels était considérable, l'offre insuffisante, les perspectives à long terme sur la disponibilité de la ressource inquiétantes. Dix ans après, nous avons constaté lors des entretiens un renversement de la perception concernant les logiciels spécifiques.

Ce changement de perception s'articule principalement autour de la constatation, commune à l'ensemble des personnes rencontrées, de l'impact des progiciels.

1.2.1 L'impact de la diffusion des progiciels sur la perception de l'activité

Nous avons demandé à nos interlocuteurs dans quelle mesure ils ont adopté des progiciels, s'ils ont une politique claire concernant ces produits et quels sont les domaines applicatifs concernés. Leurs réponses attestent de la mise en oeuvre de démarches systématiques de recherche et d'utilisation de progiciels.

Dans toutes les entreprises étudiées, les dirigeants mènent une politique volontariste en faveur de la sélection et de l'usage de progiciels : *"A chaque fois, systématiquement, on pose la question, et pourquoi voulez-vous faire cela vous-mêmes, pourquoi n'achèterions nous pas*

¹. *Electronic Data Interface*, ou encore Echange de Données Informatisé : lien électronique entre les clients ou les fournisseurs et l'entreprise.

une solution toute faite ?" (J1), ou bien : "Chaque fois qu'une solution existe, qu'elle a été développée par d'autres, on choisit l'achat de progiciel ; les gens qui ont fait le produit sont plus compétents que nous et notre objectif c'est d'être le meilleur utilisateur d'une application de ce type, bien plus que de la développer nous-mêmes". (F)

Nos interlocuteurs informaticiens reconnaissent l'incitation forte faite en faveur de ces produits par la direction générale et les utilisateurs : *"Le facteur mode joue, l'incitation est forte à choisir des progiciels" (B) ; ils se montrent parfois plus réticents : "Il y a une volonté progiciel mais elle est difficile à appliquer à cause de nos spécificités". (E2)*

Les progiciels sont introduits lors du remplacement d'anciennes applications spécifiques devenues obsolètes, comme dans le cas de deux mises en place de SAP² qui nous ont été décrites :

"Nos nouvelles applications sont à base de progiciel, nous sommes en train de mettre en place SAP. C'est une stratégie, nous avons développé des applications nous-mêmes avant" (G) et "Nous sommes actuellement dans un processus qui vise à remplacer au fur et à mesure des renouvellements nos systèmes propres dans un esprit commun par des logiciels SAP. Cela se fait progressivement, il y a une volonté de choix d'un progiciel. On a voulu créer un esprit division international. L'esprit d'intégration autour du progiciel est très marqué". (E1)

Fait intéressant, les progiciels ont fait leur apparition même dans le domaine d'applications jugées stratégiques. Buchowicz (1991) avait identifié l'importance stratégique de l'application comme le premier critère de sélection entre un progiciel et un développement spécifique interne. Quelques années après, nos observations contredisent ce résultat et montrent une évolution des comportements. Neuf entreprises sur les douze de notre échantillon nous ont ainsi affirmé considérer systématiquement l'option progiciel **y compris dans des domaines autrefois couverts par des applications maison et touchant directement au coeur du métier de l'organisation**. Dans sept cas sur les neuf, les équipes informatiques internes ont

². La gamme SAP est composée d'un ensemble de progiciels intégrés très paramétrables destinés aux entreprises du secteur industriel (gestion des achats, stocks, gestion de production, comptabilité générale et analytique ..).

mis en place des solutions mixtes, de type progiciel plus développement spécifique complémentaire. Elles jugent ces solutions avantageuses dans la mesure où le progiciel couvre la plus grande partie des besoins et où le spécifique ne représente que 20 à 30% du budget global. Dans quatre cas sur les sept, le vendeur du progiciel, bénéficiant de la connaissance du produit, a été jugé comme le mieux placé pour réaliser les développements spécifiques complémentaires (mais pas les interfaces).

Depuis plus d'une dizaine d'années, le discours sur les progiciels annonçait la diffusion de plus en plus étendue de ces produits ; cependant il était toujours assorti de limites sur la gamme de problèmes couverts par ces produits ou sur l'incapacité de ceux-ci à prendre en compte les problèmes spécifiques de beaucoup d'entreprises. Que l'ensemble des entreprises que nous avons consultées nous déclarent les adopter - ou envisager de les adopter à court terme - y compris pour la majorité d'entre elles concernant des applications stratégiques nous semble révélateur de l'évolution considérable qui s'est produite, d'une part dans les produits eux mêmes, qui ont beaucoup progressé, et d'autre part dans les mentalités des responsables d'entreprise et des informaticiens internes, qui ont levé une grande partie de leur réticence initiale envers ces produits. Utiliser un progiciel pourrait devenir aux yeux des dirigeants un moyen de réduire le risque, une assurance de rapidité, voire même dans certains cas la possibilité d'améliorer l'organisation interne en important les procédures et modes de fonctionnement dérivés du produit : *"On a fait le bilan dans certains cas entre développements propres et progiciels et aujourd'hui on reconnaît chez nous que l'on a eu quelques succès de développements propres, mais on a eu aussi quelques échecs et c'est un risque que l'on ne veut pas courir. On a fait le constat suivant, c'est que les coûts entre progiciel et spécifique sont à peu près équivalents, et le résultat du développement interne est mieux adapté, donc il donne meilleure satisfaction ; par contre il a un délai de mise en oeuvre qui est en général beaucoup plus long que le progiciel. De plus le progiciel a sa contrainte que tout le monde connaît qui est d'imposer une organisation préalable, et en fait comme très souvent ce que l'on recherche c'est une amélioration de l'organisation, le*

progiciel est aussi l'occasion de le faire. C'est plus facile de revoir l'organisation avec un progiciel qui impose des choses qu'avec un spécifique qui colle à l'existant". (E1)

La diffusion des progiciels réduit le volume d'activité du développement spécifique et change ainsi les conditions de rentabilité de l'investissement dans des équipes internes. Fait peut-être plus déterminant encore, elle **semble changer la perception des dirigeants sur l'évolution à long terme de la demande de l'entreprise en logiciels spécifiques.**

Ce changement est très sensible dans neuf entreprises sur les douze étudiées. Pendant longtemps, la demande à long terme a pu être supposée croissante, ce qui justifiait des investissements internes. La perception paraît maintenant inversée : *"A partir du moment où il y a plus de packages, moins de développement, ce sont les SSII qui développent des packages, la population d'informaticiens change, les grandes équipes de développement, c'est quelque chose qui se termine. On sait que dans ce domaine là il y a des développements en cours mais qu'après il n'y en aura plus, qu'à terme on va avoir un problème". (F)*

1.2.2 Un investissement discontinu

Un autre élément pourrait jouer sur la perception de la demande à long terme : dans cinq entreprises, nos interlocuteurs ont exprimé l'idée d'un investissement discontinu dans l'équipement en logiciels, procédant par sauts relativement éloignés dans le temps (cinq ans, dix ans pour les plus optimistes).

Cette idée s'appuie sur plusieurs constatations. Tout d'abord, même si "tout n'est pas encore informatisé" dans l'entreprise, les principales fonctions sont maintenant automatisées de façon satisfaisante ; la longue période continue de premier équipement est achevée. Ensuite, les responsables informatiques ont pu constater la difficulté de procéder par adaptations et améliorations par "ajout de couches successives" dans des systèmes très intégrés dans lesquels la modification ou l'adjonction d'une application pose des problèmes complexes d'interfaces.

Ces cinq entreprises achèvent actuellement la refonte de leurs systèmes opérationnels autour de nouveaux outils (EDI, systèmes de gestion automatisée des flux dans le site de

production, progiciels intégrateurs) ; elles ont dû, pour prendre en compte l'évolution de leurs marchés (juste-à-temps, changement du mode de relation et des communications client-fournisseur) mettre en oeuvre rapidement le remplacement de pans entiers de leur système d'information. Leur impression *a posteriori* sur cette phase intense de rééquipement, souvent menée dans des délais très rapides, est plutôt positive : même si l'investissement est lourd et concentré sur une période, il s'avère plus facile de remplacer les systèmes existants en une seule fois par un ensemble large et intégré que de procéder par étapes successives et de gérer de longues périodes de transition ; elles espèrent rentabiliser leurs investissements sur cinq à dix ans.

Ces entreprises envisagent la suite de leur informatisation comme une succession de **grands cycles pluriannuels faisant se succéder une phase d'investissement, suivie d'une phase de pause intermédiaire**, centrée sur l'exploitation et la maintenance quotidienne des systèmes, pendant laquelle la fonction de développement est réduite, ce jusqu'au démarrage de la nouvelle phase d'investissement suivante, etc. Dans ce schéma, l'activité de développement d'applications est sujette à des variations du niveau de production extrêmement sensibles. **A l'extrême, elle n'est plus considérée comme faisant partie des activités pérennes de l'entreprise.** Dans l'une des entreprises étudiées, le responsable informatique nous explique ainsi : *"J'ai été embauché au moment du nouveau schéma directeur ; le contrat moral était de deux ans. L'idée était que dans deux ans, les applications seraient en place et que l'informatique ne constituait pas une fonction structurelle de l'entreprise"*.(H)

Dans quelles conditions ce schéma est-il praticable et généralisable ? Il nous faut souligner ici le caractère prospectif de ce mode d'organisation. Dans la pratique, aucune de ces entreprises n'a atteint la phase de pause dans les investissements. Le responsable informatique engagé sur un contrat moral de deux ans est dans son poste depuis cinq ans, au cours desquels il a dû gérer d'autres adaptations imprévues, dues aux changements de structure de l'organisation (achat et cession de filiales) et mettre en oeuvre de nouveaux chantiers concernant d'autres activités.

Pour conclure sur la perception de l'activité de développement, deux points essentiels nous semblent devoir être retenus : d'une part, conformément à l'hypothèse retenue dans le choix des secteurs, les entreprises étudiées considèrent bien l'activité de développement comme une **activité de support non directement stratégique** ; d'autre part, ces mêmes entreprises entretiennent une **interrogation sur la pérennité de l'activité de développement** au sein de l'organisation, qui pourrait à terme soit être considérablement réduite du fait des progiciels et fonctionnerait sur des rythmes intermittents. Au total, on obtient une perception de l'activité de développement qui la classe parmi les activités secondaires de l'entreprise, candidates selon Quinn, Doorley et Paquette (1990) à l'externalisation.

Nous allons maintenant étudier quels sont les objectifs assignés à l'activité de développement d'applications. Nous en avons identifié deux, présents dans l'ensemble des organisations étudiées : développer la synergie et limiter les coûts.

II. L'objectif de synergie

"La première exigence de la direction générale concernant l'informatique et les nouveaux développements, c'est de coller parfaitement à la stratégie commerciale et industrielle de la société". (K)

"L'objectif de notre nouveau directeur informatique, c'est non pas de faire marcher la mécanique mais de s'assurer qu'au niveau organisation, organisation de production, achats, etc., on a une synergie de l'informatique avec l'entreprise qui fait que par rapport à l'ensemble de ce qu'il faut apporter dans ces activités qui sont assez diversifiées on investit des sommes importantes mais qui contribueront à l'amélioration de l'efficacité." (E1)

La recherche systématique d'une synergie avec le métier de l'organisation n'a dans son principe rien d'étonnant et correspond à l'objectif naturel de l'informatique depuis les débuts de sa diffusion dans les entreprises : aider celles-ci à mieux exercer leur métier en fournissant des informations et en automatisant des traitements.

Dans la pratique en revanche, et comme nous l'avons souligné dans le premier chapitre, de nombreux exemples de dysfonctionnements ont pu être constatés, le plus préoccupant concernant sans doute la fixation et le respect des objectifs assignés à l'informatique : la spécificité technique de la tâche, le degré élevé de savoir-faire et de connaissance exigés ont conduit dans certains cas les informaticiens à privilégier des buts qui leur sont propres et qui sont liés à leur statut de professionnel ; de leur côté, les dirigeants ont pendant longtemps délaissé la maîtrise d'une activité pour laquelle ils se sentaient incompetents.

Les résultats des enquêtes de Applegate et al. (1992) et de Feeny (1992) semblaient indiquer un rapprochement entre les deux groupes d'acteurs, susceptible d'améliorer la fixation et le respect d'objectifs managériaux. Nos observations confirment dans une large part cette évolution. La convergence des objectifs, l'alignement sur le métier transparaissent au travers de deux dimensions principales :

- le développement de "l'esprit de service",
- la recherche d'une plus grande réactivité et flexibilité,

2.1 Le développement de l'esprit de service

Le concept de service, la référence aux objectifs généraux de l'organisation transparaissent dans le discours de la majorité de nos interlocuteurs : *"Les opérationnels ont de plus en plus besoin d'outils informatique pour les aider à faire leur métier. Je m'occupe de cela"*. (A3)

Cette nouvelle orientation se traduit également par une évolution volontaire du langage interne : dans six entreprises sur les douze étudiées, les responsables des études délaissent le terme classique "d'utilisateur" pour parler de "client interne" ; l'importance symbolique de cette évolution peut être comparée à celle de l'administration quand elle se découvre des "clients" parmi les "usagers"³.

³. Sur ce point précis, on lira avec intérêt l'analyse faite par F. Chevalier : "La modernisation du service public : gros plan sur une démarche qualité et management participatif", in Qualité en mouvement, n° 10, février 1993.

2.2 La recherche de la réactivité

La deuxième manifestation de cet ajustement réel de l'informatique sur le métier de l'entreprise est la prise en compte par les informaticiens des impératifs de réactivité auxquels sont confrontés les opérationnels et dont dépend souvent la survie même de l'entreprise. L'accélération des processus industriels, commerciaux, financiers, se traduit par une pression très forte des opérationnels pour que l'informatique suive leur rythme, accompagne leurs évolutions, fasse preuve de la même réactivité qui leur est demandée à l'extérieur.

Dans la définition des objectifs et des plannings de développement, les impératifs de réactivité passent *avant* les problèmes de capacité de production ou les difficultés techniques. Dans toutes les entreprises sauf une, dont la production correspond à des cycles longs, on nous a expliqué que la rapidité de mise en oeuvre des nouveaux systèmes devient prioritaire ; nous en citons ici deux exemples.

Dans un premier cas, le directeur financier de l'entreprise nous explique : *"Nous réagissons très vite avec une structure très légère. Il y a trois ou quatre ans, nous avons 116 sociétés consolidées. Nous avons entrepris une régionalisation, un regroupement des entreprises, fusionné des sociétés, basculé sur d'autres systèmes administratifs. Dans le même temps, nous avons décidé de mettre en chantier des applications de gestion commerciale, de gestion de production usine et de logistique automatisée qui sont très liées et prioritaires pour nous"* (11). L'impact sur les systèmes d'informations administratif, technique et commercial de ce groupe est considérable ; le nombre et la diversité des chantiers à mettre en route dépasse de loin les capacités internes. Cependant, du côté des informaticiens, la réponse est claire : *"Nous n'avons pas de projets en attente ; on démarre tout, dès que la décision est prise"*. (12)

C'est l'appel à la sous-traitance qui permet la réactivité ; on comprend ici le poids important des facteurs opérationnels (ajustement charge/capacités, délais) déjà identifiés dans l'analyse quantitative.

Un deuxième exemple nous est fourni par le secrétaire général d'un autre groupe : *"Nous avons créé un GIE en commun avec un concurrent pour assurer la distribution de nos*

produits et nous avons eu un point sérieux de désaccord sur l'organisation commerciale. Nous pensions qu'il fallait absolument réorganiser les réseaux de distribution. Il y avait une très forte résistance de notre partenaire, il ne disait pas non, philosophiquement, mais au niveau informatique il disait attendez, ce n'est pas si simple, il faut étudier ceci, mesurer cela, on en a au moins pour 18 mois... Nous on a dit ce n'est pas comme cela que l'on travaille, on dira à nos informaticiens, messieurs, vous serez prêts tel jour. Donc on a mis trois types sur le truc, on leur a dit vous serez prêts le tant, peut-être pas totalement, mais vous serez prêts à démarrer une facturation avec la nouvelle organisation commerciale, quitte à réserver les problèmes annexes pour pouvoir fonctionner. Du coup, ce qui n'était pas possible avant 18 mois, on l'a fait. On a effectivement démarré le jour dit". (J)

On peut noter dans ces deux exemples avec quelle **fermeté** est exprimée la demande de la direction générale et des directions opérationnelles envers leur informatique. Dans le premier cas, après la décision du comité de direction, l'informatique n'avait d'autre choix que s'exécuter et de trouver des solutions techniques appropriées. De l'aveu du responsable financier, l'objectif de démarrage était "non négociable" dès lors qu'il était stratégiquement décidé : *"La direction générale est très fortement impliquée dans l'évolution de l'informatique. Elle impose d'obtenir des résultats à date précise, c'est un exemple de notre hyperréactivité". (I1)*

Dans le deuxième cas, on observe la même résolution de la part du secrétaire général : **l'argument technique** (complexité, délais de réalisation des études...) est repoussé comme non pertinent ; il **ne doit pas faire obstacle à la décision politique et stratégique**. Ces exemples, au même titre que d'autres observations convergentes que nous avons faites au cours de notre recherche terrain, nous font penser que l'opacité technique est un rempart de plus en plus fragile derrière lequel pourraient s'abriter des informaticiens davantage soucieux de respecter des impératifs techniques de production que de s'adapter en permanence aux évolutions de l'environnement et aux demandes des opérationnels. Ce que traduit l'un de nos interlocuteurs informaticiens, de façon assez abrupte : *"Il n'y a d'informatique que dans la mesure où il y a un besoin utilisateur à satisfaire... L'informatique n'a pas d'autres besoins que ceux générés par les priorités qui lui sont assignées". (F)*

Notre interprétation de ces situations est que le **discours technologique n'est plus dominant face au discours managérial**. La pression des impératifs de réactivité semble remettre en question le **mode de gestion de l'activité**, et les habitudes méthodologiques. Dans six entreprises sur les douze étudiées, les informaticiens que nous avons rencontrés témoignent ainsi d'une évolution de la gestion des développements qui s'écarte, sous la pression des impératifs de réactivité, des habitudes méthodologiques traditionnelles, axées sur une vision planificatrice à long terme incarnée par le schéma directeur.

"On a fait un plan informatique qui indique un niveau de dépense à quatre, cinq ans et indique que l'on va travailler sur tel et tel domaine. Mais il y a tellement d'impondérables auxquels on n'avait pas pensé qui arrivent qu'on ne peut pas l'appliquer réellement. Dans un mois, on fusionne les sociétés en une seule entité juridique. Les systèmes doivent suivre, ce n'était pas prévu, on va faire cela de juin à décembre. Cela nous arrive comme cela, d'un seul coup". (C1)

La part du "non prévu" dans l'activité augmente dans des proportions importantes : *"Cette année, la répartition est 60% en budget prévu, 40% en non prévu".(E2)* Un responsable informatique, ancien d'Arthur Andersen (et donc un habitué des méthodologies longues et détaillées !) nous a décrit ainsi la nouvelle organisation informatique de son groupe : *"On ne fait plus de schéma directeur. Le schéma directeur est implicite, en toile de fond, on l'a en tête, mais on ne l'a pas formalisé sous forme d'un document prospectif comportant les infrastructures, les aspects applicatifs, etc... On le voit se dessiner par rapport à une certaine vision du métier qu'on est en train de développer et qui mûrit dans les esprits des opérationnels et de la direction générale. Derrière cela, il y a des dynamiques qui se créent sur les différents domaines. Les systèmes d'information accompagnent, c'est une vision plus pointilliste qu'une approche verrouillée, couvrant l'intégralité du secteur. Dans notre métier la visibilité est à six mois. La notion de moyen terme, 5 ans, n'a pas de sens pour nous. Nous nous imposons de faire des projets à effet tunnel limité de 3 mois, sinon cela n'a pas de sens". (H)*

Ces témoignages illustrent le **besoin d'adaptation continue** de l'informatique au métier de l'entreprise, adaptation gérée à court terme, souvent dans l'urgence, qui a pour effet d'obliger les informaticiens à modifier leur cadre de référence. Le schéma directeur à cinq ou dix ans, souvent décidé et accepté par la direction générale en fonction d'impératifs techniques qu'elle ne maîtrisait pas fournissait aux informaticiens une référence stable ; l'objectif était "de mettre en oeuvre le schéma directeur". Avec la demande croissante de réactivité, cette référence disparaît ou est remise en cause dans la moitié des cas étudiés et les objectifs doivent être constamment réajustés. Enfin, rappelons que dans toutes les entreprises de notre échantillon sauf une, la réduction des délais de développement des nouvelles applications est devenue une priorité.

2.3 L'impact sur les choix d'externalisation : le maintien en interne de l'étape d'analyse du besoin

La recherche quantitative a montré la dominance de la ressource interne dans l'étape d'analyse du besoin, ce pour la très grande majorité des projets. Les témoignages de nos interlocuteurs permettent d'éclairer les motivations de ce choix.

Un de nos interlocuteurs nous explique ainsi : *"Le principe de base est que pour tout ce qui est la définition des développements applicatifs spécifiques on touche véritablement là au coeur du métier, on a besoin d'une bonne connaissance du business et des gens, des interlocuteurs dans les directions. Donc c'est quelque chose qui est confié aux informaticiens internes. Et après, quand on arrive aux phases techniques, on adjoint soit des internes, soit on fait appel à des ressources extérieures... Il n'y a pas de sous-traitance en phase initiale."* (K)

En ce sens, le maintien en interne de l'étape d'analyse du besoin semble bien répondre à des préoccupations stratégiques : la conjonction des connaissances du **métier** tel qu'il est exercé dans l'entreprise, des **hommes** de l'organisation, et de la **technologie** que possèdent les informaticiens internes s'affirmerait ainsi comme une compétence de base critique pour l'entreprise, qui s'exerce dans les étapes amont du projet (définition des objectifs, analyse du

besoin) et qui ne semble pas pouvoir être externalisée sans risque de perte d'efficacité stratégique.

En revanche, aucune des entreprises concernées n'a émis de telles réserves concernant l'externalisation des phases de conception technique et de réalisation. En particulier, elles ne semblent pas ressentir comme une nécessité de réaliser elles-mêmes leurs projets stratégiques une fois l'analyse des besoins terminés. Comme nous l'indique le secrétaire général d'un des groupes étudiés : *"Dans les applications, il y a ce qui nous est propre et que l'on ne trouvera pas sur le marché et on est les mieux placés pour savoir ce que l'on veut. Dans chaque métier, il y a une partie que l'on doit maîtriser en interne. Mais cela ne veut pas dire que l'on doit tout faire en interne. On doit le maîtriser... Quand j'ai fait faire à l'extérieur un développement et que je le possède, c'est exactement la même chose pour moi en termes de maîtrise de l'application que si je l'avais fait développer en interne"*. (J) C'est bien l'étape de définition du besoin qui apparaît comme une compétence de base stratégique pour l'entreprise. La conception et la réalisation pourraient elles être sous-traitées sans mettre en péril les objectifs de l'organisation.

Cette spécialisation selon la tâche (analyse du besoin en interne, phases aval éventuellement sous-traitées) se traduit dans l'évolution des organigrammes des services étudiés. A cet égard, les entreprises que nous avons observées présentent les caractéristiques suivantes (figure 9.1.) :

- le **responsable des applications** (ou, selon les appellations, le "responsable études") et les **responsables de domaine** (le domaine recouvre soit une spécialisation fonctionnelle - GPAO, applications administratives, applications commerciales ...- soit une spécialisation par activité de l'entreprise) sont systématiquement des **internes**,

- selon les entreprises et les situations, les **chefs de projets** peuvent être internes ou externes, avec quand même une forte **dominance d'internes** (à noter que dans certains cas, le "responsable de domaine" joue directement le rôle de chef de projet),

- les **analystes** et **programmeurs** sont souvent en majorité des **externes**, présents physiquement dans l'entreprise (régie) ou non (forfait).

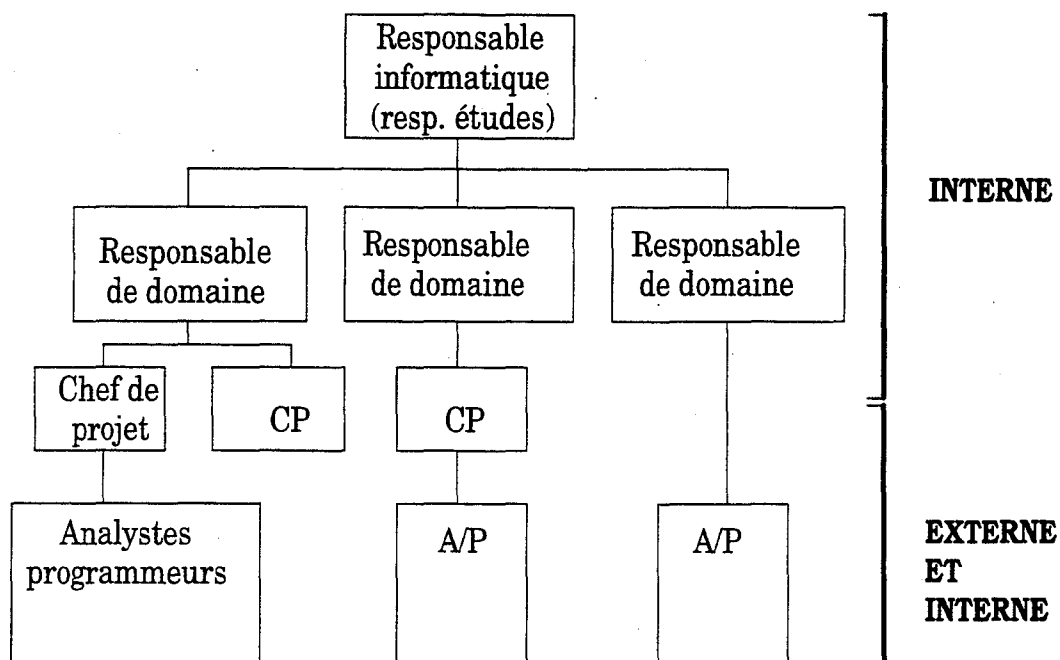


Figure 9.1 L'évolution de l'organigramme des services études

Cette organisation permet de garantir la "compétence architecturale" évoquée par Venkatesan (1992). Le responsable de domaine et le chef de projet comprennent les besoins du client interne et sont capables d'élaborer la demande en termes techniques⁴. Au total, le constat que nous avons pu faire est celui d'une majorité de directions informatiques très **conscientes des enjeux liés au métier** et souhaitant jouer un rôle de conseil et de médiation entre les opérationnels et la technologie informatique :

"Il y a une interaction entre la direction générale, les utilisateurs, l'informatique. Il y a des projets déclenchés par les directions utilisateurs, d'autres par la direction informatique. Actuellement le projet commercial est un projet davantage poussé par l'informatique que par les utilisateurs. Ils ont un système qui marche et ils ne sont pas capables de formuler une demande aussi vaste, aussi complexe que celle-là. C'est un métier d'imaginer à l'avance que le système d'information actuel a des limites, que les technologies permettent d'envisager des choses impossibles il y a 5 ans. C'est la sensibilité aux limites et aux opportunités techniques qui pousse à une meilleure réflexion stratégique." (B)

⁴. A noter cependant qu'il reste le problème du contrôle de la prestation, sujet sur lequel nous reviendrons dans la troisième section de ce chapitre.

Nous interprétons ces observations comme une remise en cause de la vision d'une informatique isolée, gérée par des experts sur des critères essentiellement techniques pour la remplacer par une **approche plus coopérative et plus flexible**, pilotée en fonction des objectifs à long terme et des opportunités à court terme du métier de l'organisation. L'informaticien interne joue alors le rôle de relais, de point de contact entre l'organisation, centrée sur son métier, et les sociétés de services informatiques, spécialistes de la technologie.

Dans l'une des entreprises étudiées, cette logique a été menée à son terme : *"Pour ce qui me concerne, nous ne développons rien. J'ai 6 personnes dans mon domaine, ce sont tous des chefs de projet et ce sont des spécialistes des différentes activités de l'entreprise. Ils assurent toute la partie recueil d'informations, spécifications avec les utilisateurs et ensuite ils font un appel d'offres et ils confient la réalisation à l'extérieur"*. (A3)

Il faut souligner cependant que le domaine d'activité en question est relativement récent (quelques années) et n'a donc pas eu à gérer d'historique, que ce soit sur le plan des applications (pas d'anciennes applications à maintenir) ou des hommes (pas d'équipe d'analystes/programmeurs). En revanche, ce domaine peut être considéré comme stratégique pour l'entreprise ; les applications développées appartiennent à la catégorie "développements innovants" de Ward. L'entreprise considère cependant que l'existence en interne d'un noyau d'informaticiens responsable à la fois du contact avec les responsables opérationnels et de l'interface avec les SSII extérieures pour les étapes de réalisation garantit la bonne mise en oeuvre de ses objectifs stratégiques. Le rôle technique de ces informaticiens demeure dans la mesure où ils jouent le rôle d'expert et garantissent la bonne exécution des travaux ; ce rôle est cependant doublé d'un rôle relationnel fort, en liaison avec les opérationnels et en cohérence avec les objectifs managériaux de l'organisation. Le parti-pris de ne rien programmer en interne interdit la focalisation des informaticiens sur le seul plan technique ; il fait passer au premier plan le rôle relationnel.

Ce cas, qui est extrême dans notre échantillon, est intéressant car il montre qu'il est possible d'envisager des solutions radicales et systématiques d'externalisation des étapes "aval" du

développement. La phase critique du processus est l'analyse du besoin ; une des conditions du succès de l'externalisation semble être alors que cette étape soit bien menée et maîtrisée en interne ⁵.

III. L'objectif de maîtrise des coûts

Synergie, oui. Mais pas à n'importe quel prix :

"On est en permanence coincés entre une demande forte d'utilisateurs, de maintenance, d'évolution, de changement et d'assistance et puis une pression très forte sur nos coûts qui fait que l'on réduit nos effectifs, donc c'est un conflit permanent entre une demande forte et une capacité réduite." (B)

"L'informatique contribue de façon annexe à nos facteurs clés de succès ; on est obligé de passer par là... mon sentiment c'est que dans la tête de beaucoup de dirigeants du groupe, l'informatique n'est pas stratégique, c'est plutôt une "utilité", on en a besoin, elle coûte cher..." (G)

"La responsabilité en informatique veut dire la transparence des coûts. Et la transparence des coûts, c'est d'expliquer comment chaque année on va réduire les coûts de fonctionnement et comment on va avoir des investissements minimum et bien ciblés pour répondre à des besoins de management, investissements qui devront être orchestrés et arbitrés en fonction des priorités des utilisateurs du management et non pas en fonction des priorités de l'informatique." (F)

La tension sur les budgets informatiques est exprimée avec force par tous nos interlocuteurs ; exacerbée par un contexte de crise grave, en particulier dans l'industrie, la volonté exprimée par la direction générale de restreindre - ou au minimum de stabiliser - les coûts de l'entreprise est manifeste dans de nombreux cas. Le budget informatique dans son ensemble ne fait pas exception : on retrouve ici l'idée d'une certaine banalisation de

⁵ Une autre condition est le contrôle des forfaits d'externalisation, que nous détaillons dans la troisième section de ce chapitre.

l'informatique, qui n'est plus une ressource bénéficiant d'un statut à part dans l'organisation, mais doit au contraire suivre les réductions budgétaires générales :

"Notre environnement en termes de volume et de marché a baissé de 12 à 20% depuis deux ans, on a licencié 1200 personnes sur un an, l'informatique ne fait pas exception." (H)

"L'informatique coûte cher, trop cher (...) on a une pression très forte sur les coûts ; on a eu une progression des budgets informatiques ces dernières années, mais on travaille sur des hypothèses de réduction qui accompagnent une réduction globale du secteur." (B)

"La division continue à faire du profit, mais dans un marché difficile qui est en récession. L'entreprise est en plan de licenciement permanent depuis des années. L'évolution du personnel informatique s'oriente vers la réduction. Même si en fait les effectifs utilisateurs dont nous nous occupons ont remonté parce que des divisions ont été regroupées, nous, à l'informatique, on a diminué." (E2)

L'impact de cette tension des budgets se fait sentir sur les modes d'utilisation des ressources : concernant l'appel à l'extérieur, toutes les entreprises recherchent la flexibilité de la ressource, avec une gestion très serrée des ajustements charge / capacité. Deux entreprises nous ont même expliqué avoir par le passé cessé totalement de faire appel à l'extérieur pour des raisons d'économies immédiates. En interne, c'est le refus d'investissements supplémentaires qui est le phénomène le plus remarquable.

3.1 Recherche de la flexibilité de la ressource

Les entreprises souhaitent jouer à fond le jeu de la flexibilité dans leur recours à la ressource extérieure. Tous nos interlocuteurs mettent en avant l'externalisation d'une partie des ressources de développement comme moyen de variabiliser les coûts :

"La partie externe est à géométrie variable." (C3)

"On est dans une logique de variabilisation des coûts. Chaque année on fait un budget base zéro et on se repose la question de la justification des effectifs internes et externes ;

jusqu'à présent on a toujours rejustifié les effectifs internes. Le principe, c'est d'avoir une partie variable importante, et donc on a une logique d'appel à la sous-traitance." (F)

L'utilisation de ressources extérieures apparaît ainsi très fluctuante dans le temps. Tous nos interlocuteurs sans exception insistent sur le caractère provisoire de la répartition interne / externe de leurs effectifs ("*actuellement..*") : "*actuellement, dans mon service, j'ai 22 internes et 8 extérieurs. Il y a également 2 extérieurs sur des forfaits*" (B) ou bien : "*actuellement, on a 20 personnes en interne et une trentaine d'externes*" (A4) ou encore : "*En ce moment nous avons 6 personnes en régie, mais elles restent sur des périodes limitées, 9 mois, un an*". (G) L'impression qui se dégage est celle d'une forte variation des effectifs extérieurs d'une année à l'autre, voire d'un mois à l'autre, selon le nombre et la taille des projets en cours.

Ainsi, l'appel à du personnel en régie doit pouvoir être déclenché à court terme, pour une durée précise et limitée : "*chez nous, la régie n'est pas gérée à long terme, mais projet par projet, étape par étape. Le chef de projet, au moment de ses phases de design sait qu'il est à 15 jours, un mois du développement où en gros interviennent les SSII ; il lance une consultation. La commande est passée pour la durée de la mission, exprimée en jours homme*". (K)

La logique de flexibilité dans l'appel à la ressource consiste également à ajuster le volume de production de l'activité de développement au volume global d'activité de l'entreprise. Les efforts d'informatisation menés par les grandes entreprises dans les années 70 et 80 leur ont permis d'automatiser les principales fonctions ; en période de récession, elles cherchent à rentabiliser les investissements engagés précédemment en prolongeant le plus possible la durée de vie des systèmes actuels. Dans sept entreprises, nos interlocuteurs ont évoqué l'existence de projets "en attente de décision" ; l'impression qui se dégage est que seuls les projets indispensables - soit parce qu'ils contribuent à améliorer la synergie, soit parce que l'obsolescence des systèmes actuels contraint à leur remplacement - sont effectivement mis en chantier.

En situation de crise sévère, les entreprises décident la réduction de la régie afin de réaliser des économies immédiates. Deux entreprises de notre échantillon ont même évoqué la suppression de la régie :

"Au départ, nous avons beaucoup d'analystes internes et peu de programmeurs mais pour supprimer la sous-traitance je veux utiliser mes ressources dans tous les secteurs et avoir la possibilité de faire aussi la programmation en interne pour des raisons budgétaires". (G)

"Notre division fait du profit mais dans un marché difficile qui est en diminution. Notre activité diminue régulièrement depuis 10 ans. Il y a 4/5 ans, on m'a dit stop. On m'a dit à partir du mois de mars, il ne faut plus de régie. Tout arrêter. Très rapidement, les gens sont partis. Je n'ai plus droit à la régie, parce que c'est assimilé à du travail temporaire et comme on a des plans de licenciements dans la division..". (E2)

Ces réactions expliquent sans doute en partie la chute des prestations de régie que l'on a pu constater ces derniers mois, tant en volume qu'en prix unitaire. La ressource extérieure joue le rôle de "soupape" de sécurité en période de restriction budgétaire. De même, le refus de réaliser en interne tout investissement supplémentaire semble ouvrir la voie à l'externalisation.

3.2 Le refus d'investissements supplémentaires.

Dans dix entreprises sur les douze étudiées, nos interlocuteurs ont souligné la volonté actuelle de leur direction générale de **ne pas investir en capital humain interne**, en tous cas pas pour les tâches d'analyse et de programmation⁶ : *"on ne veut pas avoir d'effectifs permanents pléthoriques, donc il y a un certain nombre de personnes qui travaillent pour nous et qui nous rejoignent le temps d'un projet"* (F) ou bien : *"on fait en interne sauf s'il faut recruter"*. (C3)

L'appel à un renfort extérieur était courant sur les gros projets correspondant à de fortes pointes d'activités mais il semble se radicaliser : *"Il n'est pas question d'augmenter la taille du*

⁶. Dans les deux autres cas, il a été procédé au recrutement de spécialistes de langages de développement nouveaux et mal maîtrisés en interne. Par ailleurs, la situation est différente pour des profils de type responsable de domaine ou chef de projet, pour lesquels les entreprises nous ont fait à quatre reprises état de recrutements récents.

service" (I2); "Dès que l'on dépasse trois mois hommes en général on sous-traite parce que à ce moment là cela veut dire recruter quelqu'un...". (J)

Les ressources extérieures peuvent ainsi représenter une part importante de la prise en charge des besoins : "on a sous-traité l'an dernier pour 80 mois / homme de développement, soit 47% de nos besoins" (H) ou encore : "mon service études, c'est 20 internes et 20 prestataires extérieurs".(D)

Une première explication de ces comportements peut être cherchée dans le contexte de crise actuel, peu favorable aux investissements. Nous penchons quant à nous pour un changement structurel, motivé en particulier par l'évolution des modes de production et sur lequel nous reviendrons dans la deuxième section de ce chapitre.

IV. Conclusion sur les aspects stratégiques

L'observation des situations dans les douze entreprises étudiées nous permet de mieux comprendre la prise en compte des facteurs stratégiques dans les choix d'externalisation de l'activité de développement d'applications.

L'étude théorique avait mis en lumière le lien entre l'importance stratégique accordée à une activité et l'accomplissement de cette activité au sein de l'entreprise. Nous avons élaboré et testé deux hypothèses sur le lien entre l'importance stratégique et les comportements d'externalisation :

- Notre première hypothèse était que les **projets** stratégiques, considérés comme une ressource de base de l'organisation, seraient réalisés en interne alors que d'autres projets, moins stratégiques, seraient externalisés. Cependant les résultats de l'étude quantitative n'ont montré sur les soixante projets étudiés aucune variation significative des comportements selon ce critère : que les projets soient déclarés stratégiques ou non, l'analyse des besoins est effectuée dans la très grande majorité des cas en interne ; la conception et la réalisation sont quant à elles également susceptibles d'être externalisées. L'analyse qualitative renforce ce résultat ; il semble

que les entreprises n'émettent pas de réserve au développement à l'extérieur de projets pourtant jugés stratégiques. Notre première hypothèse n'est donc pas confirmée⁷.

- Notre seconde hypothèse prenait en compte l'importance stratégique des différentes étapes au sein du processus. Selon ce critère, seule l'étape d'analyse du besoin justifiait d'être internalisée.

Les résultats de l'approche qualitative confirment ceux de l'analyse quantitative et sont cohérents avec cette seconde hypothèse. Pour expliquer leurs choix d'externalisation sur un plan stratégique, nos interlocuteurs se réfèrent en effet à l'importance stratégique des tâches effectuées dans les différentes étapes du processus. L'enjeu prioritaire est le développement d'une synergie entre l'informatique et le reste de l'entreprise et le maintien en interne des étapes critiques. Cet enjeu se joue principalement lors de l'étape d'analyse du besoin. A ce stade, nos interlocuteurs déclarent préférer les informaticiens internes, dans la mesure où, outre leur savoir informatique, ils connaissent le métier tel qu'il est pratiqué dans l'organisation et ont su développer des relations avec les principaux responsables utilisateurs.

Nous résumerons ce point en disant que, pour nos interlocuteurs :

- parmi les trois étapes étudiées (analyse du besoin, conception, réalisation), seule la première semble être considérée comme une "ressource de base" de l'organisation ainsi qu'une étape critique du processus, susceptible d'être maintenue en interne pour des raisons stratégiques,

- pour cette étape, l'avantage distinctif des informaticiens internes se fonde davantage sur leurs compétences managériales (connaissance du métier et des hommes) que sur des compétences techniques également partagées par les fournisseurs extérieurs.

Au-delà du test de ces deux hypothèses, l'analyse qualitative permet d'enrichir notre compréhension de l'attitude des entreprises vis-à-vis de l'informatique dans des secteurs où elle

⁷. La situation serait peut-être différente pour des entreprises où l'informatique est très stratégique et où se posent des problèmes de protection des nouvelles applications contre d'éventuels plagiats de la part des concurrents, ce qui pourrait conduire à internaliser également la conception et la production.

n'est pas directement stratégique. Des incertitudes sur la pérennité de l'activité de développement d'applications spécifiques ont été exprimées à plusieurs reprises par nos interlocuteurs, nourries par un usage croissant des progiciels. Le double objectif annoncé par l'ensemble des entreprises étudiées (assurer la synergie tout en limitant les coûts) nous paraît être quant à lui la manifestation du souhait de "rentabiliser" l'activité de développement tout en la mettant directement au service de l'entreprise.

Section 2 - Les contraintes industrielles : volumes et flux de production, technologie, difficultés d'intégration

L'approche stratégique nous a fourni des éléments de réponse concernant le maintien en interne de l'étape d'analyse des besoins. En revanche, elle ne permet pas d'éclairer directement les choix pour les étapes de conception et de réalisation, auxquelles nos interlocuteurs ne semblent pas attacher d'enjeu stratégique suffisant pour justifier leur internalisation systématique.

Nous allons dans cette section étudier dans quelle mesure l'approche industrielle peut nous aider à comprendre les comportements observés. Nous avons identifié à l'issue de l'analyse théorique trois types de facteurs industriels :

- les facteurs liés à l'évolution des volumes et flux de production (pérennité de la demande, ajustements charge / capacité, délais),
 - les changements technologiques,
- ces deux séries de facteurs sont à priori favorables à l'externalisation,
- les difficultés d'intégration des nouveaux développements aux systèmes existants, facteur *a priori* défavorable à l'externalisation.

I. Facteur favorable à l'externalisation : l'évolution des volumes et flux de production

1.1 Une perception aiguë de l'irrégularité des flux de production

Nous avons fait état dans la première section de ce chapitre de deux évolutions ayant un lien direct avec les volumes et flux de production.

La première concerne la perception globale de l'activité et des volumes de production à long terme : nous avons vu que la quasi-totalité de nos interlocuteurs anticipe une baisse sérieuse du nombre de développements spécifiques ; certains doutent même de la pérennité à

long terme de la demande de l'entreprise ou envisagent de longs cycles successifs faisant alterner des phases d'équipement intenses et des périodes de sous-activité.

La seconde évolution touche le mode de gestion de l'activité ; tous nos interlocuteurs font état du souhait d'accélération des processus et de la nécessité d'une bonne réactivité de l'informatique pour atteindre l'objectif de synergie avec le reste de l'entreprise et répondre aux contraintes de l'environnement. Le cadre rigide du schéma directeur est apparu plusieurs fois bouleversé, tant pour des raisons d'évolution de l'environnement et du marché de l'entreprise que sous la pression des contraintes budgétaires ; les demandes des utilisateurs sont exprimées souvent à court terme, dans l'urgence, avec des délais de réalisation serrés.

Au total, ces deux évolutions se conjuguent pour former un environnement de production caractérisé par une fluctuation importante des volumes d'activité : fluctuation à court terme, due à l'impact des variations de l'environnement; fluctuation à moyen et long terme, où l'on est passé du contexte de pénurie des années 80 au risque d'avoir à gérer bientôt le sous-emploi des équipes internes.

Quelles sont les solutions possibles pour gérer l'irrégularité des flux ? Les entreprises que nous avons étudiées ont mis en oeuvre de façon séparée ou conjointe trois types de stratégies qui jouent sur les choix d'externalisation :

- la première consiste à tenter de rendre la ressource interne plus flexible, c'est-à-dire à étudier la reconversion des analystes et programmeurs dans d'autres activités,
- la deuxième, adoptée par deux entreprises, est d'élargir la base de clientèle de l'informatique interne par une politique de centralisation des développements au sein du groupe,
- la troisième stratégie, la plus largement utilisée par ces entreprises, est une politique d'évitement de capacité ⁸

⁸ Selon l'expression de Marchesnay, cité par Reix (1979b)

1.2 Les tentatives difficiles de reconversion du personnel interne

L'avenir des développeurs internes (analystes et programmeurs) est une préoccupation commune à la plupart des entreprises avec lesquelles nous avons travaillé. La situation à court ou moyen terme ne semble pas poser de difficultés réelles : ils peuvent être affectés à des tâches de maintenance des applications spécifiques ; par ailleurs l'intégration des systèmes (réalisation des interfaces) devient avec la multiplication des progiciels un chantier majeur, qui nécessite la connaissance du système informatique existant.

A plus long terme cependant, elles sont confrontées à trois problèmes majeurs.

Le premier concerne **l'affectation de ces analystes et programmeurs aux nouveaux développements**. La difficulté réside essentiellement dans la maîtrise de la technologie.

Le second problème est la **gestion des carrières** de ces personnels, en admettant qu'ils restent dans leur domaine de spécialité, l'informatique. Le responsable informatique de l'un des groupes étudiés explique ainsi les raisons de sa préférence pour l'emploi d'équipes extérieures : *"La part du développement dans les entreprises diminue, ici comme dans d'autres, donc il serait irréaliste même dans des endroits où l'on a temporairement des développements importants à faire de maintenir un nombre de salariés importants et de ne pas pouvoir proposer des évolutions de carrière normales. Donc on utilise des renforts temporaires."* (F)

Le troisième problème est celui de la **reconversion**. Parmi les entreprises étudiées, nombreuses sont celles dans lesquelles la politique de gestion du personnel est centrée sur un engagement à long terme (on fait sa carrière dans le groupe et on en sort à la retraite). La règle est donc celle d'une certaine flexibilité et de mutations internes qui permettent l'ajustement des ressources en fonction de la demande. A cet égard, le personnel informaticien s'avère particulièrement rigide. Reix (1979b) nous donne une clé d'interprétation en distinguant dans chaque emploi la part d'apprentissage spécifique (non transférable) et celle d'apprentissage général. Plus la part d'apprentissage général est élevée, plus l'individu pourra être aisément reconverti vers d'autres fonctions. A l'inverse, les emplois spécialisés sont ceux où la ressource est la moins flexible. Les postes des informaticiens se distinguent - à tout niveau, mais ce

d'autant plus qu'une part importante de l'activité est consacrée à des tâches techniques - par un apprentissage spécifique très élevé.

Un de nos interlocuteurs, gestionnaire de formation, en a fait l'expérience difficile quand il a accepté de prendre la responsabilité des applications de son groupe : *"Moi je ne suis pas du tout informaticien à la base et même avec une formation ce n'est pas évident. Pendant un an, c'est même très très pénible..."* (K) Le poids de la spécialisation est même si fort qu'un individu ayant une formation générale mais occupant depuis quelques années un poste technique en informatique voit son image confondue avec celle d'un spécialiste incapable de se reconverter vers un apprentissage général : *"J'ai deux personnes, deux ingénieurs avec une formation générale d'électromécanique [cette formation correspond au métier de l'entreprise] qui voulaient partir de l'informatique, ils n'ont pas pu partir parce que les autres divisions ont préféré embaucher. Ils m'ont dit il faut que l'on embauche des gens du métier. Et moi j'ai des gens qui, même s'ils ont au départ une formation de base, après cinq ans d'informatique, sont devenus des informaticiens et ne sont plus considérés que comme des informaticiens".* (E2)

Dans l'une des entreprises étudiées, où la décision a été prise de sous-traiter systématiquement la programmation, des essais de reconversion ont été tentés pour faire évoluer les programmeurs vers des tâches plus généralistes d'encadrement. Les résultats sont décevants : *"On s'aperçoit que c'est très difficile de changer de métier. Ceux qui aiment la programmation, ils veulent continuer à faire de la programmation, même s'ils disent on voudrait bien être chef de projet, on s'aperçoit qu'ils reviennent, qu'il faut qu'ils touchent les programmes. C'est très difficile. On a mis en place des plans de formation, on a essayé, on s'aperçoit que c'est difficile, très très difficile, faire changer quelqu'un c'est dur."* (K)

Cette difficulté de reconversion du personnel informaticien interne (analystes et programmeurs) a conduit le responsable informatique de l'un des groupes étudiés à engager une **politique volontaire de désinvestissement**, par le biais d'opérations de tierce maintenance. Lors de l'entretien, il s'exprime avec réticence sur ce point sensible, en demandant notamment de cesser l'enregistrement. Il explique avoir choisi de mettre en tierce maintenance

certaines vieux applicatifs à espérance de vie limitée (2, 3 ans) dans l'objectif principal de reclasser du personnel interne à l'extérieur. Les salariés du groupe sont devenus les salariés de la société de service avec laquelle a été conclu le contrat de maintenance. Il explique la motivation de l'opération par l'impossibilité d'utiliser ces salariés à terme dans le même type de fonction (les applicatifs vont disparaître), l'impossibilité de les faire évoluer vers d'autres fonctions informatiques plus proches de l'utilisateur et enfin par le sentiment que les analystes et programmeurs forment une population peu intégrée à l'entreprise et de plus difficile à gérer, évoquant ainsi le problème du contrôle sur lequel nous reviendrons dans la troisième section de ce chapitre.

Nous n'avons rencontré qu'un seul exemple de ce type d'externalisation ayant pour objectif le reclassement du personnel interne à l'extérieur. Cette pratique est très courante dans certains secteurs des services. La presse informatique fait également écho de ces contrats de tierce maintenance incluant des clauses de reprise du personnel interne. Pour autant, les onze autres entreprises étudiées ne semblent pas prêtes à choisir ce type de solution et nous ont déclaré vouloir maintenir l'emploi des équipes internes.

1.3 Les stratégies d'élargissement de la base de clientèle

Du fait des fluctuations importantes d'activité, le maintien au sein de l'entreprise d'équipes d'analystes et de programmeurs doit s'appuyer, pour être économiquement justifié, sur une base de clientèle (d'utilisateurs) de plus en plus grande. C'est la solution retenue dans deux des groupes étudiés : dans le premier cas, on a effectué un regroupement des équipes de développement d'applications servant différentes activités au sein d'un service centralisé, ce afin de préserver le maintien d'économies d'échelle. Dans le second cas, la direction du groupe envisage une centralisation et une spécialisation à l'échelle internationale : la filiale française pourrait devenir le fournisseur en logiciels spécifiques pour l'ensemble des filiales européennes, quitte à voir ses effectifs renforcés par une partie des équipes informatiques locales.

Cette stratégie n'est accessible qu'aux grands groupes ; elle répond à une logique de spécialisation menée en interne, à l'échelle du groupe, et destinée à garantir des volumes de

production suffisants. En l'absence de tels volumes, les entreprises s'engagent dans des stratégies d'évitement de capacité.

1.4 Les stratégies d'évitement de capacité

Sans aller jusqu'à pratiquer des politiques volontaires de désinvestissement et de reclassement du personnel à l'extérieur de l'entreprise, les entreprises, confrontées à une rigidité importante du personnel qui limite les possibilités de reconversion, tentent de préserver leur flexibilité à long terme en engageant des stratégies d'évitement de capacité.

Externaliser la production permet en effet d'ajuster la permanence de la ressource à celle de l'emploi. Il suffit de considérer pour s'en convaincre les durées de vie très différentes des trois types de contrats que nous avons étudiés : avoir du personnel interne correspond à un investissement sur le long terme (et même le très long terme car dans les groupes que nous avons observés, où le personnel est en règle générale assez stable, recruter un informaticien implique de gérer sa carrière pendant vingt, trente voire quarante ans) ; à l'opposé, le forfait est par définition borné dans le temps, souvent à relativement court terme (les délais se comptent en mois) ; entre les deux, la régie présente des formes multiples, avec dans certains cas des engagements se prolongeant sur plusieurs années mais également des contrats précaires à durée très limitée.

En situation d'incertitude sur la pérennité de la demande en développement d'applications, éviter d'investir est une stratégie de minimisation du risque. C'est l'interprétation que nous donnons - avec les aspects technologiques que nous allons développer dans la section suivante - au refus d'investissement que nous avons constaté dans la quasi-totalité de ces entreprises, qui ne peut ainsi être réduit à un phénomène conjoncturel de crise.

II. Le poids de la technologie : la logique de spécialisation entre différents acteurs du marché

L'étude quantitative menée sur les 60 projets nous a permis de mettre en évidence une très forte externalisation des phases techniques du projet (programmation mais aussi, quoi que dans une moindre mesure, conception). Le poids de la technologie apparaît également très clairement dans le discours de nos interlocuteurs, qui confirme largement notre hypothèse selon laquelle l'évolution technologique actuelle est un facteur favorisant l'externalisation. Deux propositions se dégagent dans ce discours, que nous allons étudier tour à tour :

- l'externalisation permet de mieux maîtriser une technologie de développement de plus en plus complexe et multiforme,
- l'investissement interne dans la technologie doit rester limité.

2.1 L'externalisation permet de mieux maîtriser une technologie de développement de plus en plus complexe et multiforme

L'analyse quantitative a montré le lien entre la nouveauté technologique d'un projet et son externalisation.

La complexité de la technologie, l'impossibilité d'en maîtriser toutes les formes en interne et l'intérêt d'une spécialisation des acteurs sur le marché se dégagent également très clairement des entretiens. Citons à titre d'exemple les réactions de quelques uns de nos interlocuteurs, qui traduisent bien l'opinion selon laquelle l'entreprise ne peut prétendre, dans un domaine aussi hétérogène, à atteindre un niveau de compétence équivalent à celui de sociétés spécialisées :

"Les métiers des systèmes d'information, c'est une foultitude de métiers, on ne peut pas être bons partout. Aujourd'hui le développement, il y a des technologies tellement différentes, ça bouge tous les jours, donc là, on n'est pas organisé, on n'est pas structuré pour gérer des programmeurs, pour industrialiser la production, il y a des sociétés qui sont beaucoup plus fortes que nous." (A1)

"On ne peut pas avoir d'équipes de réalisateurs qui soient bons dans tous les domaines techniques. On a des applications qui sont en Unix, sous des réseaux Novell, il faudrait recruter quelqu'un qui connaisse tous les nouveaux langages ... tandis que là, on prend à l'extérieur 3 personnes spécialistes Oracle / Unix." (D)

L'hétérogénéité des techniques renforce la part d'apprentissage spécifique et rend, à notre avis, plus difficile encore le problème de reconversion des équipes techniques que nous avons déjà souligné. Le développement quasi universel en Cobol est remplacé par différents outils, langages, environnements spécialisés ; du fait de l'évolution technologique, la ressource interne perd de sa flexibilité, même en restant dans le cadre restreint de l'activité de développement : *"Former un expert sur un domaine technique pointu cela peut prendre deux ans et nous n'avons pas le temps d'attendre deux ans (...) On a intérêt à passer par des entreprises extérieures, à aller chercher les spécialistes là où ils sont quand on en a besoin et ne pas tout faire en interne"* et, à propos d'un projet à forte composante technique : *"Je ne vois aucun d'intérêt à faire en interne la réalisation. Parce que il y a tout un tas de spécialités et les SSII sont structurées pour ça. Pour nous c'est un projet qui dure quatorze mois avec sept ou huit personnes de la SSII pour développer. Après on ne saurait plus quoi en faire, ce sont des spécialistes." (A3)*

Pour réaliser les nouveaux développements, les entreprises se sentent dans de nombreux cas moins compétentes que les prestataires spécialisés. Ce sentiment d'incompétence concerne la maîtrise pratique des nouveaux outils :

"A l'extérieur, je recherche une spécificité : par exemple, j'ai démarré des projets client / serveur sur micro, j'ai une personne en régie qui nous aide parce qu'elle connaît bien l'environnement SQL Windows Oracle. Ou alors j'ai un problème Novell, je trouve un spécialiste" (E2).

"On travaille avec des SSII spécialisées en Ingres ; on a besoin d'expertise technique pour le développement " (H).

"L'entreprise a fait le choix du départemental avec UNIX sous Oracle et n'a pas de qualification dans ce domaine. Donc elle s'appuie énormément sur la sous-traitance parce qu'elle n'a pas les compétences techniques." (L)

La difficulté technique est une voie d'entrée naturelle vers une externalisation radicale, où le fournisseur prend en charge la responsabilité du projet : *"Sur un certain nombre de petits projets, bien identifiés, liés à une technique particulière, télécoms, en outre on peut avoir un développement externe sous-traité forfaitisé."*(F) On retrouve ici le lien identifié dans l'étude quantitative entre le caractère technique des projets et le choix de contrats au forfait.

2.2 L'investissement interne dans la technologie doit rester limité

Dans ces conditions, les entreprises envisagent-elles de faire l'effort d'apprentissage des nouvelles technologies de développement ? Nos observations indiquent qu'elles n'ont pas complètement renoncé à faire cet investissement, mais qu'elles entendent le **restreindre à quelques domaines limités et directement opérationnels dans l'entreprise**. Dans plusieurs cas est exprimée une incertitude sur la rentabilité de choix technologiques audacieux, ainsi que la crainte de surenchère technologique. La technologie doit rester secondaire et répondre à des objectifs précis en liaison directe avec le métier de l'entreprise ; l'expertise interne n'est nécessaire que dans des cas très ciblés :

"Il n'y a pas d'intérêt à regarder telle et telle chose uniquement pour se cultiver. Par exemple on réfléchit beaucoup aux réseaux Novell, mais uniquement parce que c'est ce que nous avons... Ce n'est pas nous qui arrivons sauf sur des cas très spécifiques aux limites de la technologie. Quand nous utilisons une technologie, elle a forcément été banalisée par quelqu'un de plus important que nous. Et l'objectif, c'est d'être dans les meilleurs standards, déjà banalisés, du marché. Nous sommes des suiveurs. Nous sommes très contents quand nous sommes les meilleurs utilisateurs d'une excellente technique déjà banalisée." (F)

"Par rapport à la technologie, notre mission est de rester pragmatique, ne pas se perdre dans des recherches fondamentales technologiques, à 10 ans... autrement dit toute recherche

technologique devrait être supportée par l'idée que ce pourrait être applicable dans tel ou tel secteur." (K)

Les entreprises que nous avons étudiées n'ont donc pas franchi le pas qui les conduirait à se reposer uniquement sur l'appel à l'extérieur pour leurs principales applications. **Elles tiennent toutes à maintenir en interne un niveau de compétence technologique.**

Pour six d'entre elles, l'objectif est de pouvoir assurer elles-mêmes la maintenance : *"Dans la mesure du possible, on essaie d'intégrer la connaissance, parce que on ne va pas confier la maintenance et j'ai tout intérêt à acquérir le maximum de connaissance", (E2)* ou encore : *"On fait appel à l'extérieur dans les phases de développement et puis après on essaie de rebasculer vers du personnel interne pour assurer la maintenance des applicatifs". (B)*

Pour les six autres, la connaissance de la technologie répond à un objectif de contrôle (et non plus de réalisation) des tâches ultérieures : *"On essaie de choisir des standards du marché. Et en général, on met quelqu'un du service sur le coup, de façon à ce qu'il suive cela et qu'il puisse s'y retrouver, de façon à avoir l'option après, même si on ne l'exerce pas, quand il faut faire des modifications, soit de faire en interne, soit de faire faire, par celui là ou par un autre. Encore faut-il pour cela que l'on connaisse suffisamment ce que l'on a acheté" (J)* ou encore : *"En principe, non, on ne fera pas nous-mêmes la maintenance. Mais il faut avoir des analystes / programmeurs au courant ne serait-ce que pour pouvoir être l'intermédiaire entre l'utilisateur et puis le fournisseur au niveau de l'évolution de spécifications soit chez nous, soit chez le prestataire." (F)*

III. Limite à l'externalisation : l'intégration des nouvelles applications aux systèmes existants

Si les entreprises reconnaissent volontiers envisager l'externalisation du fait de fluctuations importantes du volume d'activité de développement et de difficultés à gérer l'évolution technologique, elles mettent en revanche en avant un obstacle sérieux à une externalisation radicale, qui est la difficulté d'intégration des nouveaux développements aux systèmes d'information existants.

L'obstacle est à la fois technique et financier. Technique, parce que l'apprentissage de systèmes existants souvent incomplètement documentés et fruits de couches successives est à la fois long et difficile, les risques d'erreur importants. Financier, car les analystes et programmeurs de l'entreprise ayant déjà investi dans la connaissance des systèmes existants bénéficient d'un net avantage en termes de coûts comparés : *"Il y a des parties de nos applicatifs où il n'y a vraiment que les gens d'ici qui peuvent faire une analyse parce que c'est tellement imbriqué que quelqu'un de l'extérieur, ce serait beaucoup trop coûteux qu'il aille rentrer là dedans et ce serait même dangereux parce que c'est extrêmement sensible."* (A4)

Cette limite à l'externalisation est très sensible dans cinq entreprises de notre échantillon, dont les systèmes d'information se caractérisent par la présence de grandes applications spécifiques datant de plus d'une dizaine d'années.

Les responsables du développement de ces entreprises nous ont expliqué qu'ils emploient de préférence du personnel interne (ou des salariés en régie longue durée) pour la réalisation des interfaces et des maintenances évolutives. On retrouve ici un des résultats de la recherche quantitative, à savoir que plus le logiciel est intégré aux systèmes existants, moins l'entreprise a tendance à externaliser sa production. La pénalité encourue par le fournisseur, qui doit investir dans une connaissance préalable du site d'implantation, le rend moins susceptible de bénéficier d'un avantage comparatif de coût ; la notion de risque semble également ici importante.

L'impact de cette limite à l'externalisation doit cependant être nuancé ; l'analyse qualitative montre ainsi que l'on ne peut pas généraliser ce résultat. En particulier, les opérations de refonte globale d'un pan entier du système d'information, que nous avons déjà décrites et qui ont été mises en oeuvre dans cinq des entreprises étudiées, permettent de contourner l'obstacle de l'intégration au systèmes existants.

IV. Conclusion sur les aspects industriels

Cette section nous a permis de dépeindre la nouvelle logique de spécialisation industrielle qui semble se dégager de l'évolution actuelle.

Notre interprétation des situations observées nous conduit à isoler un ensemble de facteurs favorables à l'externalisation.

Le premier argument industriel en faveur de l'externalisation est l'évolution des volumes et flux de production.

Les entreprises que nous avons étudiées appartiennent à des secteurs où l'informatique n'est pas considérée comme stratégique. Elles entretiennent des incertitudes sur la **pérennité de leur propre demande** en développements spécifiques. Elles sont de plus confrontées à des variations à court terme des volumes de production, conjuguées à des contraintes sévères de **décalai** qui renforcent le poids du facteur d'**ajustement charge/capacité**. Elles cherchent à améliorer la **flexibilité de la ressource**, c'est-à-dire à pouvoir engager les ressources appropriées au moment où elles en ont besoin et à ce moment là seulement et font donc appel à la sous-traitance. On retrouve ici les résultats de la recherche quantitative quant à l'influence des facteurs opérationnels (ajustement charge/capacité, délais) sur les comportements d'externalisation. La recherche qualitative renforce ces résultats mais permet également de prendre du recul sur l'influence de ces facteurs opérationnels et de les intégrer dans une perspective à plus long terme sur l'activité de développement, marquée par une incertitude sur la pérennité.

Le second argument en faveur de l'externalisation est l'hétérogénéité croissante de la technologie de production.

L'apparition de nouvelles techniques de développement pose des problèmes de compétence interne, d'efficacité et d'efficience. Dans l'évaluation des solutions sur des projets mettant en oeuvre de nouvelles technologies, le fournisseur extérieur apparaît souvent comme le meilleur pour au moins deux critères sur trois : meilleure qualité (il a déjà développé avec les mêmes

langages, outils, concepts et a donc acquis de l'expérience) ; meilleurs délais (il dispose d'un personnel formé et disponible à court terme) ; et même souvent meilleurs coûts, quoique cet avantage soit moins fréquemment cité par nos interlocuteurs.

Des limites à l'externalisation sont également apparues. Il faut ainsi noter que toutes les entreprises de notre échantillon jugent nécessaire **d'entretenir en interne un bon niveau de compétence sur quelques technologies, très ciblées, qui sont au coeur de leur propre système d'information.**

Un autre frein important à l'externalisation est la **difficulté d'intégration des nouveaux développements au sein du système d'information existant**, car elle limite sérieusement les avantages en termes de coût dont peut bénéficier un prestataire extérieur.

Enfin, les entreprises étudiées ne recrutent plus d'analystes ou de programmeurs, mais elles ont **maintenu dans leurs postes les effectifs internes** et ne se sont pas lancées, à l'exception de l'une d'entre elles, dans des politiques de reclassement de leur personnel à l'extérieur. Le poids de la maintenance et le nombre de nouveaux projets en cours de développement leur permettent pour l'instant d'occuper largement les effectifs internes. Si le volume d'activité diminuait, elles auraient alors à pratiquer un arbitrage entre le licenciement de personnel interne et la réduction de l'appel à l'externalisation. Nous pensons que le souhait d'occuper en priorité les effectifs internes pourrait alors constituer un frein à l'externalisation.

Section 3 - Le problème du contrôle

Notre troisième grande hypothèse, dont nous allons examiner l'application dans cette section, est que les choix d'externalisation sont liés aux possibilités de pilotage du fournisseur, interne ou extérieur, par l'organisation. Selon cette approche, l'entreprise peut être amenée, sur le critère du contrôle, à adopter d'autres choix d'organisation que ceux induits par les critères stratégiques ou une logique industrielle.

Les trois types contractuels étudiés correspondent à trois situations de contrôle différentes (cf. chapitre 6) :

- l'organisation interne correspondant à des contrats personnalisés à long terme et à un type de contrôle hiérarchique,
- la régie est une forme intermédiaire (contrat relationnel) et conjugue les modes de pilotage de l'organisation interne et du marché,
- le forfait est la forme la plus proche du marché ; elle s'appuie sur une évaluation du résultat, à l'opposé des deux formes précédentes qui reposent sur l'observation et le contrôle des comportements.

La question est de savoir quelle est la performance des différents modes de contrôle pour le développement d'applications et éventuellement de déterminer quel est le mode de contrôle le plus approprié, c'est-à-dire celui qui permet d'atteindre correctement les objectifs et limite les coûts de transaction.

Nous avons donc demandé à nos interlocuteurs de nous décrire les modes de contrôle utilisés, de porter une opinion sur les résultats obtenus et de nous faire part des éventuels échecs. Cette section est organisée autour des trois types contractuels identifiés : interne, régie, forfait. Pour chacun d'entre eux, nous analysons sur le plan du contrôle de l'activité le discours de nos interlocuteurs.

I. Le pilotage de la relation interne

Est-ce l'effet de la tension actuelle très forte sur les coûts ou bien est-ce le signe d'une maturité plus grande des entreprises dans la maîtrise de l'informatique ? Toujours est-il que nous avons pu constater dans l'ensemble des organisations que nous avons étudiées la mise en oeuvre systématique de politiques de contrôle des coûts de développement.

Toutes les entreprises de notre échantillon utilisent un système de refacturation des coûts de développement. La direction informatique est considérée comme un centre de responsabilité avec une ligne de résultat. L'utilisateur se voit imputer au final le coût des prestations reçues. L'objectif de la direction informatique est d'équilibrer son budget.

Le budget est la base de la refacturation ; en cas de dépassement, il y a négociation entre les deux parties pour savoir qui est à l'origine du dépassement (modification des demandes des utilisateurs, coûts de production non respectés ou mal estimés au départ par les informaticiens), discussion et affectation au compte de résultat du responsable du dépassement. Ces procédures permettent d'améliorer la visibilité des coûts de développement ; elles sensibilisent les utilisateurs à leur consommation de ressources et incitent ceux-ci à exercer une pression à la baisse sur les coûts : *"Le budget est un contrat, établi au départ, avec des discussions un peu tendues, en disant ça vous coûte tant pour tel service. Le contrat budgétaire est passé en début d'exercice et après on roule là dessus."* (E1)

Du moins est-ce ainsi dans les principes ! Aussitôt cette règle énoncée, nos interlocuteurs nous expliquent qu'elle est extrêmement complexe à appliquer. Nous avons pu observer dans la pratique la mise en oeuvre de trois types de solutions, conduisant à des situations de contrôle différentes. Ces solutions diffèrent principalement sur trois points :

- 1 - qui est le détenteur du budget (informatique ou utilisateurs)
- 2 - quel est le mode de définition des budgets
- 3 - comment se traitent les arbitrages concernant les ajustements apparaissant en cours de développement et non prévus au budget.

1.1 La sévérité du marché : le budget comme référence unique

Premier mode de fonctionnement, la "sévérité du marché". Cinq entreprises entrent dans cette catégorie (C, F, G, H, I). Dans ces entreprises, le budget apparaît comme la référence unique sur laquelle se fonde le dialogue entre l'informatique interne et les utilisateurs.

Pour chaque nouveau projet de développement, un budget est établi, d'abord globalement, puis très précisément sur la base du cahier des charges ; il fait l'objet de discussions entre les responsables utilisateurs et les informaticiens, ce jusqu'à conclusion d'un accord. Les informaticiens s'engagent sur la base de cet accord ; le contrôle des dépenses engagées en cours de projet est très formalisé et très strict : *"Les utilisateurs ont des budgets d'investissement pour leurs applications. Pour chaque nouvelle application, l'utilisateur doit remplir une "fiche d'initiative" qui est la demande d'autorisation à la direction générale du groupe. En cas de dépassement, il faut remplir une nouvelle fiche d'initiative et la faire accepter."* (C) . **Il faut noter que dans cette configuration, c'est l'utilisateur qui a le budget de développement et non l'informatique.**

Le budget correspond à une **enveloppe globale, établie en début de projet**. Une fois le budget établi, les informaticiens ont **interdiction de dépasser les montants prévus** et se voient imputer quasi systématiquement les dépassements. Dans ces entreprises, les responsables utilisateurs déclarent : *"Nous n'acceptons pas d'avenants importants de la part de notre informatique"* (I1), ce à quoi répondent en écho les informaticiens : *"On n'a pas de dépassement, cela n'existe pas"* (I2). L'idée à la base de ce fonctionnement est d'appliquer en interne et avec la même sévérité les contraintes associées aux relations de marché.

Ces entreprises utilisent des règles très strictes d'engagement de dépenses. Sur les projets, le responsable informatique n'a pas le droit de consommer de ressources au-delà des montants alloués. L'élaboration du budget informatique est elle même très surveillée. Dans un cas, on utilise la méthode du "BBZ" (budget base zéro, toutes les dépenses et tous les effectifs doivent

être rejustifiés chaque année). Dans un autre, la notion de budget informatique a quasiment disparu : *"Dans le processus budgétaire il y a des coûts de fonctionnement minimum que l'on cherche à réduire et des projets identifiés dont le budget appartient aux utilisateurs. L'informatique n'a pas de budget propre, au sens où il n'y a pas de département méthode ou recherche et développement. Un investissement en GPAO⁹ , n'est pas dans le budget informatique ; il y a des coûts informatiques liés, mais l'investissement est dans le budget de l'usine. Quand je discute avec des prestataires, on me dit, oui, dans votre budget informatique, pour la part télécom, il y aurait bien une petite place pour... Non. Tout ce qui ne correspond pas à un besoin exprimé par un client interne, sur lequel il y aura une réponse qui fera l'objet d'une demande d'autorisation d'investissement... Non. Ce n'est pas notre budget."* (F)

D'après nos interlocuteurs, la solution est généralement efficace en termes de coûts, c'est-à-dire que l'enveloppe initiale du budget est respectée.

Ces cinq entreprises présentent des caractéristiques communes entre elles et différentes de celles des autres entreprises étudiées, ces caractéristiques touchent l'organisation de l'activité de développement et le choix des projets :

- peu d'adaptations en cours de développement.

Une fois l'analyse des besoins validée par les deux parties, la règle est le respect strict du cahier des charges. Les aménagements et améliorations éventuels sont reportés à une phase ultérieure, qui devra faire l'objet d'un nouveau budget,

- une politique progiciel très accentuée.

Ces cinq entreprises sont celles qui se sont le plus engagées dans des politiques systématiques d'utilisation de progiciel, pour des raisons de coût,

- un choix de projets donnant la priorité aux applications d'automatisation de processus formalisés, dont la rentabilité à court terme peut être établie.

⁹ . Gestion de Production Assistée par Ordinateur.

Notre interprétation de ces situations est celle d'un lien entre la rigidité budgétaire et ces différents éléments.

La rareté des adaptations en cours de développement est à notre avis une conséquence directe de la contrainte budgétaire. Les informaticiens doivent s'engager auprès des utilisateurs sur une facturation sur la base du cahier des charges. Or, il est souvent difficile d'estimer toutes les conséquences d'une modification dans les fonctionnalités de départ ; c'est un risque que les informaticiens, se sentant en position de faiblesse dans la négociation avec les utilisateurs, refusent d'encourir. Il tendent donc à s'opposer à toute modification du contrat de départ par les utilisateurs, de peur de provoquer des dépassements qui risquent de rester à leur charge. Pour les utilisateurs, la contrainte de devoir demander un budget supplémentaire est également un frein aux demandes de modification.

De plus, une très forte contrainte budgétaire nous semble inciter les informaticiens à privilégier le choix de certains processus de développement au détriment d'autres ; elle peut conduire en particulier au rejet de modèles de type "développement évolutif"¹⁰, qui peuvent prendre en charge une expression des besoins au départ incertaine et incomplète de la part des utilisateurs mais qui comportent trop d'ajustements pour entrer dans un cadre budgétaire strict. Dans la majorité des cas, c'est donc un modèle séquentiel de développement (cycle en cascade) qui est adopté ; ce modèle a pour principal inconvénient sa rigidité ; celle-ci est encore accentuée par la contrainte budgétaire. Au total, l'entreprise est conduite à ne plus se lancer dans des développements innovants et aux contours mal définis et à privilégier les applications d'automatisation de processus formalisés dont la rentabilité à court terme peut être établie.

Il nous semble également qu'un tel système focalise l'attention davantage sur les *coûts* des prestations que sur les prestations elles-mêmes. La relation client-fournisseur porte principalement sur la valeur (le montant inscrit au budget) et n'intègre pas toujours de garantie suffisante sur le contenu ou la qualité. Un risque de dysfonctionnement est alors de respecter le budget... au détriment de la qualité du service : *"Il y a des choses que l'on ne fait pas. On ajuste par la quantité"* (H). Nous pensons que les règles de gestion incitent dans une certaine

¹⁰. Comme le modèle de spécification opérationnelle décrit par Agresti (1986), cf. chapitre 3.

mesure à adopter ce type de comportement, dans la mesure où l'arbitrage entre une amélioration possible de la qualité du logiciel fourni et le respect du coût initial est dans la majorité des cas tranché en faveur de la limitation des coûts. L'arbitrage en faveur des logiciels pour des raisons de coût nous semble aller dans le même sens. Sauf dans des cas très précis, le développement spécifique est perçu comme un luxe non justifié.

Enfin, dans leur discours, les informaticiens de ces cinq entreprises se situent d'emblée en prestataire de services techniques chargés de la maîtrise d'oeuvre du projet et se comparent volontiers à une SSII externe. Ce point de vue n'est pas surprenant si l'on considère qu'ils se trouvent dans une situation de quasi-marché où chaque acteur a tendance pour protéger ses intérêts à jouer un jeu individuel ; la coopération informaticiens-utilisateurs est limitée aux aspects opérationnels en liaison directe avec les projets (élaboration du cahier des charges, recette...). La synergie de l'informatique avec le reste de l'entreprise se limite à fournir les outils adaptés aux demandes formulées par les utilisateurs, au moindre coût.

Les responsables informatiques confrontés à ce type de contraintes semblent ainsi privilégier le pilotage des aspects techniques de leur tâche et investir peu dans les aspects managériaux, à l'exception de la négociation des budgets avec les utilisateurs. L'initiative sur les évolutions fonctionnelles de l'informatique vient principalement des directions opérationnelles ; le rôle de conseil de l'informatique interne est peu développé, sauf en ce qui concerne la réalisation technique de projets initiés par les responsables utilisateurs.

Au total, le bilan de ce mode d'ajustement qui consiste à tenter de recréer en interne un mode de pilotage proche de celui du marché, essentiellement régulé par le prix des prestations, apparaît mitigé. L'efficacité du contrôle des coûts est obtenue ; en contrepartie, il semble que l'on perde la capacité d'adaptation coopérative *a priori* caractéristique du mode interne. Il faut noter qu'avec ce mode d'organisation on s'éloigne radicalement de l'archétype de la hiérarchie décrit par Williamson. Les obligations des informaticiens par rapport au budget n'ont rien "d'élastique", le pardon des erreurs et des fautes est loin d'être la norme.

On retrouve ici des éléments de la théorie des prix de cession interne (Ardoin et al., 1986), appliquée avec un objectif de rentabilité des différentes unités. Les utilisateurs ont la délégation

des budgets de développement et le choix de solutions alternatives ; l'informatique est soumise à la concurrence. La concurrence peut être *externe* (possibilité de confier au forfait de gros projets à une SSII extérieure) mais aussi *interne* au groupe (service informatique d'une autre filiale du groupe). L'essentiel est que les informaticiens aient le sentiment de ne pas être la source unique ; en cas d'échec sur un projet (mesurable principalement par le dépassement du budget ou le non-respect des délais) la sanction de l'appel à une autre source peut être appliquée. En cas de recours à un fournisseur extérieur, c'est le même mode contractuel axé sur le respect du budget initial qui est adopté. Une autre condition de la mise en oeuvre de cette rigueur budgétaire concerne la nature des produits concernés par les prix de cession internes : ainsi, le fait que ces entreprises s'orientent vers des solutions de type progiciel est favorable à la mise en oeuvre de règles d'arbitrage strictes. Enfin, la faible importance stratégique accordée à l'activité semble également déterminante.

Premier mode de fonctionnement interne : la sévérité du budget	
Conditions de mise en oeuvre :	
<ul style="list-style-type: none"> • l'entreprise se limite aux projets bien structurés et dont la rentabilité peut être prouvée • forte volonté des dirigeants ; existence de solutions alternatives • l'utilisateur est responsable du budget des projets 	
Avantages	Inconvénients
<ul style="list-style-type: none"> • contrôle du coût des prestations - estimation <i>a priori</i> - pas ou peu de dépassements 	<ul style="list-style-type: none"> • rigidité, résistance aux adaptations en cours de processus de la part de l'informatique interne • problème du contrôle de la qualité des prestations (difficulté de définition complète du résultat a priori, étanchéité du processus) • emphase sur le rôle technique des informaticiens ; rôle managérial peu développé
Rôle joué par l'externalisation :	
L'externalisation est choisie par l'utilisateur sur un critère de coût ; elle permet également de garantir la sévérité des sanctions en cas de non-respect du budget.	

Tableau 9.1. La sévérité du marché : le budget comme référence unique

C'est dans ce groupe d'entreprises que nous avons trouvé la plus grande expression de vulnérabilité de l'informatique interne ; la visibilité des responsables informatiques sur la pérennité de leur service ne dépasse pas le laps de temps nécessaire à l'achèvement des projets en cours ou prévus à très court terme. Au-delà, ils envisagent des réductions d'effectifs, ou, - qui sait ?-, d'autres projets : les données et les réponses ne sont pas entre leurs mains ; elles dépendent de la direction générale et des utilisateurs ; ils n'ont pas le sentiment d'avoir grande influence sur ceux-ci.

1.2 L'ajustement coopératif

Certaines entreprises privilégient en revanche pour leurs développements internes un mode d'ajustement coopératif ; nous en avons trouvé deux exemples très marqués dans notre échantillon ; d'autres tentent de mettre en oeuvre ce type de fonctionnement.

Un informaticien décrit ainsi la procédure d'ajustement budgétaire dans son groupe :

"Les dépassements marchent par avenants. S'il y a modification de la demande ou si l'on découvre que c'est plus compliqué que ce que l'on avait pensé, en général en interne on discute et ça se négocie. Si on a gagné du temps, on le dira aussi. L'enveloppe qui est établie sur le plan financier n'est pas complètement définitive, c'est une enveloppe de coût pour les utilisateurs, qui leur permet d'avoir une visibilité sur leur budget informatique, mais elle n'est pas complètement verrouillée comme on peut avoir avec une SSII externe. C'est quand même beaucoup plus souple. (...) On n'a pas des forfaits avec une marge de sécurité énorme, on essaie de faire des devis qui sont au plus court, on ne se met pas la marge de sécurité de 20% pour absorber l'incertitude. Parce que l'on sait qu'effectivement, si on a vraiment un problème, et bien on en discutera et cela se passera bien." (A4)

On peut noter dans ce discours la présence des éléments caractéristiques de l'**adaptation coopérative** décrite par Barnard (1938) : l'information est partagée (en cas de dépassement comme en cas de gain de temps, les utilisateurs sont mis au courant), les utilisateurs et les informaticiens s'engagent dans un système coopératif appuyé sur des règles de **confiance**

réciroque. Le chiffrage initial établi par les informaticiens comporte peu de marge de sécurité car en cas d'ajustement la régulation pourra se faire sur le principe de la "bonne foi" des parties.

La règle est la communication d'information et la **transparence**. Dans un autre groupe ayant également adopté cette approche, le responsable informatique explique : *"Au départ, les informaticiens étaient calés sur cette règle : tant que l'on n'est pas à la fin de la phase de spécifications, je ne lâche pas de chiffre parce que, après, je ne veux pas avoir d'ennuis. Je leur ai dit, non, on va mettre les choses clairement sur la table avec les clients. Ce n'est pas possible qu'en face, ils ne sachent pas qu'ils vont en avoir potentiellement plutôt pour 2 ou plutôt pour 6 MF. Donc systématiquement, on donne un ordre de grandeur afin qu'en face ils puissent dire, est-ce que vraiment je veux mettre 6 MF là dedans, est-ce que je ne préfère pas les mettre ailleurs. Pour compenser un peu le fait que plus le projet avance, plus vous êtes engagés, vous ne revenez pas en arrière."* (K)

Il est frappant de constater l'importance que prend ici la réciprocité et l'équité de la relation entre les deux parties du contrat. Considérant qu'en phase avancée du projet l'utilisateur ("le client") est de fait en situation de dépendance vis-à-vis de son fournisseur interne, le souci du responsable informatique est de préserver l'équité de la relation en communiquant toutes les informations dans les étapes initiales ; si cette règle du jeu n'était pas respectée, le rapport de confiance initial, "l'engagement" ne serait pas possible.

Comme dans le premier groupe, la règle est le respect du budget. Cependant ici le **budget**, au lieu de correspondre à une enveloppe globale prévue dès le départ pour l'ensemble du projet **s'établit dans une démarche progressive qui laisse la place aux ajustements** : *"Pour un projet donné, on se met d'accord avec le client phase par phase : la phase de scoping, définition des frontières, cela fait tant. Vous acceptez ou vous n'acceptez pas. A partir de là, je suis engagé : si j'ai dit 200.000 francs et j'en ai pour 250.000, la différence est pour le compte de résultat de l'informatique. La direction pour laquelle je travaille ne sera débitée que de 200.000 francs. On avance comme cela dans le projet phase par phase, sous forme de devis acceptés ou pas acceptés, rediscutés, qui finissent par être fermes"*(K). Plus qu'une

obligation contractuelle, le respect du budget après chaque phase d'ajustement a ici valeur de symbole de la réciprocité de l'engagement et devient une garantie pour les ajustements futurs.

Quand elle fonctionne, cette organisation présente des avantages non négligeables : utilisateurs et informaticiens travaillent dans un mode coopératif : *"Les liens entre l'informatique et les utilisateurs sont très serrés très quotidiens, nous sommes dans un service très proche de l'exploitation et de la production, donc on a des liens quotidiens, extrêmement personnalisés, c'est une relation qui se joue sur la durée. Aujourd'hui le professionnalisme et la connaissance du métier qu'ont les informaticiens parce qu'ils sont là depuis des années et des années font que les utilisateurs sont compris à demi-mot. Ils expriment un besoin de façon synthétique et derrière les informaticiens complètent l'analyse".(A4)*

Autre avantage, utilisateurs et informaticiens ont un objectif commun qu'ils définissent ensemble. C'est dans ces deux entreprises que nous avons vu les informaticiens prendre la plus large part à la définition des objectifs stratégiques ; ils considèrent la veille technologique comme partie intégrante de leur métier et exercent une fonction de conseil auprès des utilisateurs. Dans ce contexte, la synergie réclamée par les opérationnels et les dirigeants et dont nous avons décrit les principales lignes dans la première section de ce chapitre prend tout son sens. Il ne s'agit pas seulement de fournir rapidement un service conforme à la demande au moindre coût, comme c'est le cas dans le premier groupe d'entreprises ; la synergie s'exerce dans l'élaboration en commun de la définition du "meilleur" service, adapté aux objectifs de l'entreprise.

La coopération entre utilisateurs et informaticiens se fonde sur une connaissance commune du métier de l'entreprise qui permet aux informaticiens d'interpréter les objectifs des opérationnels, de les traduire en besoins informatiques, voire parfois de jouer le rôle d'anticipation des tendances du marché et de conseil :

"Le problème dans cette activité, c'est de s'y retrouver, de comprendre le marché, comment il évolue, où il va, de comprendre où on en est, de se fixer des objectifs et d'être capable de piloter relativement facilement. Ce truc là était quelque chose qui avait été couvert très partiellement par des extractions d'information à partir de la facturation et par

des méthodes end-user computing. C'était loin du sujet. La façon dont on a défini cela avec le responsable de l'activité, c'était de dire à tout moment je veux avoir une vue sur mon business suivant n'importe quel angle d'attaque. Donc l'application a consisté à mettre à plat l'activité, conceptualiser, constituer une gigantesque base de données et puis mettre à disposition l'information de façon la plus simple possible, avec des outils d'interrogation sophistiqués du vendeur chez lui avec son Minitel". Ce qui nous frappe dans cet exemple, c'est le discours de l'informaticien. Le problème pourrait être défini de façon uniquement technique (il s'agit de la modélisation d'une base de données et de la mise en place d'un réseau) ; ce n'est pas l'optique adoptée. Au contraire, face à une demande imprécise, l'accent est mis sur le problème de management qui est perçu dans sa complexité ("*le problème, c'est de s'y retrouver, de comprendre, de savoir où va le marché, comment il évolue...*") et avec empathie ("à tout moment je veux avoir une vue sur *mon* business...").

La mise en oeuvre d'une telle organisation ne peut être envisagée qu'à certaines conditions.

Tout d'abord, le "**profil**" de l'**équipe de direction de l'informatique interne** semble déterminant : dans les deux cas étudiés, il s'agit d'individus assez jeunes, très sensibles aux aspects managériaux de leur fonction. Leur préoccupation est au moins autant de développer et d'enrichir leur connaissance du métier de l'entreprise que de gérer les aspects techniques.

Le deuxième point important à souligner est que la confiance des utilisateurs, qui est à la base de la relation avec les informaticiens internes, ne leur est pas seulement "donnée" sur la base d'un historique de bonnes relations. Elle se nourrit de manifestations très concrètes qui sont autant de **preuves** de la volonté de l'informatique interne de faire passer l'intérêt de l'entreprise avant le sien propre. Dans le mode d'organisation étudié précédemment, la facturation est globale et correspond impérativement au budget ; la "cuisine interne" de l'informatique ne concerne pas les utilisateurs. Ici, au contraire, l'informatique interne joue la transparence et fournit le plus de détails possibles sur son fonctionnement et sur le mode d'élaboration de ses coûts, en recherchant volontairement la comparaison avec les fournisseurs extérieurs : "*On a décidé de mettre l'informatique un peu sur le même rang qu'une SSII extérieure. Par exemple on publie tous les ans des tarifs internes, qui vont depuis la location*

mensuelle de tel type de micro jusqu'à l'heure de programmation ou l'heure de chef de projet." (K)

Dans cette comparaison, l'informatique interne insiste sur le long terme et sur la transparence de ses actions, à comparer avec l'opportunisme des acteurs sur le marché : *"Une des grandes tendances du client est de dire que l'informatique interne, c'est trop cher. Quand je suis arrivé en 1989, il y avait une activité dans le groupe qui ne gagnait pas beaucoup d'argent et donc tirait sur tous ses postes de dépenses. J'ai dit d'accord, sur le projet suivant, on va faire un appel d'offres extérieur. Si vous trouvez moins cher, développement et maintenance, globalement, je fais le même prix ; si vous trouvez plus cher, je facture plus cher. Et bien ils n'ont jamais fait l'appel d'offres. Ils ont très bien compris. Ils avaient fait cela une fois avec un constructeur, en 1985, qui avait fait du dumping sur le développement et les avait assassinés sur la maintenance. Au bout d'un an ils commençaient à ne pas être très contents. Ils sont venus voir l'informatique, qui leur a dit «débrouillez-vous». Après le constructeur a fermé le site sur lequel tournait l'application, il y a eu des coûts d'adaptations complémentaires qui ont été débités à l'activité." (K)*

Troisième élément saillant, c'est sur la **qualité** de la prestation et sur son adéquation parfaite au métier de l'entreprise que l'informatique interne fonde son avantage distinctif ; c'est l'ajustement continu qui permet de mettre en oeuvre cette qualité, utilisateurs et informaticiens ayant chacun le droit à l'erreur et travaillant en coopération sur le long terme pour élaborer le produit adapté : *"Pour tout nouveau projet, on demande aux utilisateurs les enjeux, on calcule un pay-back et on demande à notre client l'accord pour payer. On fait systématiquement un contrôle du budget à la fin du projet et en plus on fait avec les utilisateurs ce que l'on appelle un reappraisal du projet deux ans après. Est-ce que cela a bien permis de capturer sur le plan business ce que l'on cherchait ? Comment sont les coûts de fonctionnement par rapport à ce qui avait été prévu ? Donc la rentabilité qui était prévue de tant, de combien est-elle ? Et peut-on éventuellement corriger ?"(K)*

On trouve au travers de ces deux cas d'entreprises une confirmation de notre hypothèse selon laquelle l'adaptation coopérative, loin de pouvoir être systématiquement associée au mode interne, requiert des conditions spécifiques : ici, elle se fonde sur une communauté d'intérêts pour le métier et un langage managérial partagé, une transparence des processus et l'accent sur la qualité des prestations. Dans le cas contraire, les risques de dysfonctionnement sont réels, comme nous allons le voir avec l'analyse du dernier type de situations de contrôle.

Second mode de fonctionnement interne : l'ajustement coopératif	
<p>Conditions de mise en oeuvre : conditions d'une bonne adaptation coopérative</p> <ul style="list-style-type: none"> - profil de l'équipe d'informaticiens internes (orientation managériale) - transparence du processus pour gagner la confiance des utilisateurs - accent sur la qualité des prestations ; processus de contrôle de la qualité à long terme - découpage du budget par phases ; l'utilisateur est responsable du budget. 	
<p>Avantages</p> <ul style="list-style-type: none"> - qualité de la synergie entre l'informatique interne et le reste de l'entreprise - possibilité d'ajuster le projet en cours de processus 	<p>Inconvénients</p> <ul style="list-style-type: none"> - incertitude sur le respect du budget final
<p>Rôle joué par l'externalisation :</p> <p>La comparaison avec un fournisseur extérieur est un moyen de légitimer l'informatique interne sur le plan du coût des prestations. L'utilisateur peut choisir la source extérieure.</p>	

Tableau 9.2. L'ajustement coopératif

1.3 L'auto-contrôle de l'informatique interne

Enfin, cinq entreprises de notre échantillon ont adopté une solution privilégiant l'auto-contrôle de l'informatique. Dans cette situation,

- l'informatique est responsable du budget,
- elle refacture intégralement ses prestations aux utilisateurs en fonction de prix standards et du volume d'heures effectuées pour chaque service,

- l'utilisateur doit s'adresser à l'informatique interne pour les développements. Elle n'a le droit de s'adresser directement à l'extérieur que pour des petits projets marginaux.

Le point important à souligner nous semble ici que, quand la relation coopérative ne s'établit pas et que de plus la contrainte budgétaire est faible, **l'entreprise risque de se trouver en situation de non contrôle de son informatique interne**. C'est la situation que nous avons observée en particulier **dans une division de l'une des entreprises étudiées (E2)** et que nous allons maintenant décrire ; on y retrouve l'ensemble des dysfonctionnements signalés par exemple par Nolan (1977) et dont nous nous sommes fait écho dans le premier chapitre. D'autres entreprises de notre échantillon présentent également certains de ces dysfonctionnements, sans qu'ils aient la même portée. Notons également plusieurs de nos interlocuteurs nous ont déclaré chercher à réduire le recours au personnel interne de crainte de rencontrer les mêmes problèmes.

Les règles du jeu dans cette entreprise sont apparemment les mêmes que dans les entreprises des deux premiers groupes ; le responsable des études nous déclare : *"On publie nos résultats, on ne fait pas de bénéfice, on refacture tout, on fait un bilan que l'on diffuse aux utilisateurs et qui est transparent."*

En réalité, les différences sont profondes.

Contrairement au premier groupe ("la sévérité du budget"), les utilisateurs n'exercent pas de véritable contrôle, ni sur les budgets, ni sur le choix des projets : *"La responsabilité finale est au responsable organisation et informatique. On regarde ce qui nous semble nécessaire; on fait la liste des projets qu'il nous semble nécessaire de développer"*.

Pas de contrôle réel également de la conduite de l'informatique ou de la qualité des prestations. Ici, l'informatique interne n'est organisée ni pour évaluer de façon précise les budgets, ni pour rendre des comptes... Le budget qui est délivré par l'informatique interne est volontairement approximatif : *"Certains utilisateurs demandent combien cela va leur coûter... Mais on essaie de ne pas rentrer dans ce type de discussion, puisque, de toute façon, c'est eux*

qui paieront. Donc on donne un ordre de grandeur, mais on n'est pas organisé pour rendre compte à l'utilisateur des dépenses par la suite."

Comment traite-t-on les ajustements ? L'informaticien explique : *"Si on a décidé d'engager le projet et que l'informatique s'est trompée, pour telle ou telle raison, et bien je dirai tant pis pour eux ! [les utilisateurs]"*.

Il n'est apparemment pas troublé par le paradoxe de la situation, et ajoute : *"Avec l'expérience, on n'est pas trop mauvais tout de même concernant les évaluations. On est plus proche que l'utilisateur. Pour le projet GL¹¹, les utilisateurs avaient fait une estimation d'une dizaine de mois/homme, moi j'ai fait une estimation de 140 mois/homme et on a fini à 195. Donc on est moins mauvais qu'eux. Et on aurait pu mettre 250..."* Ce qui semble manquer ici, c'est à la fois une véritable sanction de l'erreur et l'existence même d'une autorité hiérarchique, qui ne parvient pas à s'imposer sur le service informatique.

Contrairement au second groupe ("l'ajustement coopératif"), le contrôle ne peut pas être obtenu par le biais d'une coopération volontaire entre utilisateurs et informaticiens. L'orientation de l'informatique interne est résolument technique ; à entendre certains propos de notre interlocuteur, responsables des études pour une division du groupe, l'intégration au reste de l'entreprise semble pour le moins discutable : *"L'effectif de la division ? Ah, je ne sais plus... Juste avant les vacances je l'avais bien en tête... Je ne sais plus"*. De même, il se déclare incompétent pour nous indiquer les principaux objectifs de la division.

Concernant les ajustements par rapport au budget, le responsable des études indique, avec ce qu'il faut bien appeler un certain cynisme (ou désabusement ?) : *"Il y a plein de manières de respecter un budget si on a envie de le respecter : dans la répartition tâches hommes / machine, on fait un peu plus homme et un peu moins machine. Dans ce cas là, on prend tel processus on le laisse manuel, telle procédure on la laisse manuelle, et on arrive à respecter un budget. Par contre, on peut avoir un décalage par rapport aux objectifs. Les objectifs disparaissent dans le temps. Tout change dans le temps, ce n'est pas très grave."*

¹¹ · Projet de mise en oeuvre d'une base de données comptable.

La situation décrite peut paraître extrême ; on ne peut bien heureusement pas généraliser ces exemples de dysfonctionnements à l'ensemble des situations d'auto-contrôle. Cependant, sans aller aussi loin, certaines réflexions entendues lors des entretiens expriment également des préoccupations quant au contrôle de l'informatique interne par l'informatique interne :

- sur le respect en interne des objectifs : *"quand vous êtes en interne, vous avez la direction informatique qui va s'ériger en seul juge, à la limite même de la finalité du projet, au nom de la technique, et qui va la contredire ?"* (J)

- sur celui des coûts : *"à partir du moment où vous avez 15 personnes pour faire du développement, vous avez 15 personnes que vous pouvez charger avec des projets, des dérapages, c'est l'informatique qui contrôle."* (F)

- et sur la motivation du personnel interne : *"Quand j'utilise des extérieurs, je peux agir sur la fibre commerciale. Sur des équipes internes, je n'ai aucune corde. Le fait d'avoir des moyens d'action, de changer quelqu'un de place, en interne ce n'est pas possible."* (D)

Concernant les choix d'externalisation, on peut noter que la direction informatique de cette entreprise, même si elle fait appel à des fournisseurs extérieurs, reste la seule source disponible; au contraire des deux groupes étudiés précédemment, le choix entre développement interne et sous-traitance est du ressort des informaticiens.

Ce souhait des informaticiens internes de conserver le monopole de la relation avec les fournisseurs semble plus fréquent dans de grandes structures ; nous l'avons observé plusieurs fois lors de contacts avec des responsables de grands services informatiques internes. Citons par exemple l'un de ces responsables: *"Nous souhaitons très fermement que le responsable utilisateur s'adresse à la direction informatique et que celle-ci lui dise, «cela représente tant». Que ce soit la direction informatique ou Bossard, Unilog, Cap Cesa, ou n'importe quelle autre petite boîte, ce n'est pas le problème du responsable utilisateur. Et je ne souhaite pas que ce soit son problème. Je ne souhaite surtout pas qu'il vienne chercher et mettre son nez dans le choix de la société de services."*

Au total, notre interprétation est que la situation de non contrôle apparaît comme la résultante d'une **situation d'auto-contrôle mal gérée. Les facteurs de dysfonctionnement nous semblent être :**

- **le désengagement des utilisateurs** et de la direction générale dont la seule consigne depuis plusieurs années pour la division informatique de cette division semble avoir été de limiter globalement les budgets,

- **le manque de précision et d'engagement sur les budgets,**

- **l'inexistence de règles strictes de suivi des dépenses,**

- **la position de force de l'informatique interne qui est pour l'utilisateur la seule source disponible,** même si elle fait dans un second temps appel à la sous-traitance.

1.4 Le lien entre les modes de contrôle de l'informatique interne et les comportements d'externalisation.

Au travers des trois portraits de situations que nous venons de tracer, semble se dessiner un lien entre le mode de contrôle de l'informatique interne retenu par l'entreprise et ses comportements d'externalisation.

Dans le premier cas, le contrôle est fortement axé sur les coûts. L'entreprise cherche à appliquer en interne la même sévérité que celle du marché en imposant un respect strict des budgets ; les informaticiens internes s'axent principalement sur leur rôle technique et se comportent de façon très similaire à un fournisseur extérieur ; ils peuvent dans certains cas développer des comportements opportunistes en sacrifiant le contenu de la prestation, moins visible, au respect des coûts. Placés en position de faiblesse par rapport aux utilisateurs, ils sont réticents à toute adaptation *a posteriori* du contrat initial. Dans ce cadre, l'externalisation éventuelle est décidée par la direction générale ou par les responsables opérationnels principalement sur des critères de coût ; elle permet d'exercer une pression forte sur les informaticiens internes.

Dans le second cas, la relation entre les informaticiens internes et les utilisateurs se caractérise par l'ajustement coopératif. Les deux groupes partagent une expérience commune du métier de l'organisation, les informaticiens interviennent en tant que conseil interne. Pour être crédible cependant, la bonne foi des informaticiens doit être rendue visible par des critères objectifs. La transparence des processus internes, la comparaison avec le coût de prestations extérieures et la possibilité pour les utilisateurs de choisir éventuellement un fournisseur externe servent de garantie.

Dans le troisième cas enfin, on se trouve en situation d'auto-contrôle. La direction générale et les utilisateurs se situent en retrait. L'appel à un fournisseur extérieur est décidé par l'informatique, sur des critères de production (capacité, technologie). Les utilisateurs se trouvent eux face à une source unique, ce qui les empêche d'exercer toute véritable pression. Dans cette situation, le risque nous semble être que l'informatique interne agisse comme un "bastion" technique et ne rende pas clairement compte des résultats obtenus.

Le tableau 9.3 reprend ces différents éléments.

Mode de contrôle interne	Qui décide du recours à un fournisseur extérieur ?	Objectifs généraux de l'externalisation
Sévérité des budgets (Contrôle par les utilisateurs)	La direction générale et les responsables utilisateurs	Réduire les coûts de production ; faire pression sur l'informatique interne
Ajustement coopératif	La direction générale, les responsables utilisateurs et l'informatique interne	Garantir la compétitivité de l'informatique interne ; offrir aux utilisateurs une solution alternative
Auto-contrôle informatique	L'informatique interne	Résoudre des problèmes de production

Tableau 9.3 Portrait de trois types d'organisation :

le lien entre le mode de contrôle interne et les comportements d'externalisation

Après le pilotage en interne, nous abordons maintenant un deuxième type contractuel très utilisé par les entreprises que nous avons étudiées : la régie.

II. L'externalisation en régie : les évolutions d'un mode de pilotage intermédiaire

La régie existe depuis longtemps dans le domaine informatique ; c'est encore le mode le plus utilisé pour faire appel à un fournisseur extérieur, loin devant le forfait.

Nous montrerons dans un premier temps comment la gestion traditionnelle de la régie (la "régie longue durée") permettait de qualifier ce mode intermédiaire de "quasi-interne"; nous analyserons ensuite les raisons du déclin de cette forme de régie et l'évolution vers de nouvelles pratiques, (la "régie forfaitée"), associées à de nouveaux modes de contrôle.

2.1 Un mode traditionnel "quasi-interne" : la régie longue durée

Nous l'avons vu avec la théorie des coûts de transaction, la régie est une forme de gouvernance intermédiaire, fonctionnant à la fois sur des structures de pilotage propres aux activités gérées en interne (contrats personnalisés à long terme, motivation par la hiérarchie, contrôles administratifs, adaptation coopérative) et sur des mécanismes de régulation propres au marché (incitation et régulation par le prix, concurrence entre plusieurs fournisseurs, possibilité de sanction en cas de non-respect des obligations). Si l'on adopte le point de vue de la théorie de l'agence, la régie peut même être confondue avec certains modes d'utilisation d'équipes internes, s'appuyant sur un contrôle du comportement. Elle se situe alors à l'opposé des contrats qui reposent sur un contrôle du résultat, comme le forfait et comme certains modes de fonctionnement interne que nous avons décrits ("la sévérité des budgets").

Il nous a paru intéressant d'étudier la perception de ce mode de pilotage chez nos interlocuteurs : un mode de régulation (hiérarchie, marché) domine-t-il l'autre ou s'agit-il au

contraire, sur le plan du contrôle, d'une véritable forme intermédiaire ? Peut-on assimiler régie et équipes internes ?

A l'analyse des quinze entretiens menés avec des informaticiens, la **perception des deux modes, interne et régie, semble très proche, à tel point que l'amalgame est courant**. Un responsable des études nous déclare ainsi : *"les internes, c'est ceux qui sont chez moi et dont je contrôle les travaux directement"* (A2) ; ce groupe inclut de fait à la fois les salariés de l'entreprise et du personnel en régie. Un autre nous décrit la régie comme *"une ressource longue durée, des gens qui sont là depuis trois ou quatre ans, on ne sait pas s'ils font partie de la société ou pas"* (C3). Un troisième précise : *"Mes plus anciens programmeurs sont des gens de sociétés extérieures. Je travaille depuis longtemps avec une société installée à proximité et dont j'occupe pratiquement en permanence un peu moins de la moitié des effectifs. Il y a des analystes-programmeurs chez eux qui ont travaillé pour le groupe il y a dix ans sur des projets et qui travaillent pour le groupe depuis."* (E2)

Une autre personne enfin indique : *"la régie longue durée, c'est presque pour moi du développement interne. La seule chose qui change, c'est que j'ai une facture du fournisseur au lieu d'avoir une facture interne. Mais sinon le fonctionnement est exactement pareil."* (B)

La plupart de ces discours font référence à la **durée** des contrats de régie. Beaucoup d'entreprises pratiquent (ou *pratiquaient*) une régie de longue durée, qui permet de développer une relation extrêmement personnalisée, où la relation de marché avec un fournisseur se transforme en relation de coopération avec un individu précis, intégré aux équipes internes. L'un de nos interlocuteurs résume cette perception avec la formule suivante : *"Quand je parle de régie, je parle de personnes, je ne parle pas de fournisseur"*.(B)

A tel point que l'on se demande souvent où se trouve la frontière entre l'interne et l'externe! Sur un plan juridique, elle se situe entre les salariés de l'entreprise et les fournisseurs extérieurs. La perception de nos interlocuteurs est cependant souvent différente.

Écoutons par exemple le discours de cette responsable de domaine, qui encadre une équipe de développeurs mixte (interne et régie) et fait également de temps en temps appel au forfait : *"Ces développements sont faits par nous. On prend des programmeurs en régie ; il n'y a pas de forfait là dessus"* (A4). Ici, c'est le forfait et le forfait uniquement qui est perçu comme extérieur ; la régie équivaut au mode interne.

Nous avons constaté à plusieurs reprises ce même écart de perception au cours des entretiens menés auprès d'informaticiens. Plus que la référence juridique, **c'est le mode de pilotage qui semble déterminant de la perception d'externalisation** pour ceux-ci :

- est "interne" ce qui est physiquement présent et que l'on contrôle directement (le personnel interne et celui en régie),
- est "externe" ce qui est situé en dehors des locaux de l'entreprise et dont on ne supervise pas la réalisation (le forfait).

La proximité avec les thèses de la théorie de l'agence est tout à fait frappante. Pour les responsables informatiques, - qui adoptent un point de vue de pilotage des tâches - , la grande différenciation n'est pas l'appartenance statutaire mais le type de pilotage, au comportement - englobant à la fois l'interne et la régie -, ou au résultat - concernant le forfait.

2.2 Le déclin de la régie longue durée

Ce mode traditionnel, que nos interlocuteurs désignent souvent sous le nom de "régie longue durée" tend selon eux à disparaître. Deux éléments expliquent ce déclin. Tout d'abord la régie a joué ces dernières années le rôle de "soupape de sécurité" pour les entreprises confrontées à la crise économique. Ensuite, la régie longue durée se révèle inadaptée au contexte actuel de forte évolution technologique.

2.2.1 Une réserve qui sert de "soupape de sécurité" en cas de crise

La fragmentation, voire la suppression des contrats de régie longue durée constitue un moyen facile et relativement indolore pour l'entreprise - indolore, en tous cas, comparé à des licenciements - de réduire ses coûts à court terme. La régie joue donc le rôle de réserve organisationnelle : en période de croissance, les effectifs augmentent ; en période de crise ils sont réduits pour permettre à l'entreprise de continuer d'assurer sa compétitivité, voire dans certains cas sa survie.

Le fait que le personnel en régie soit très intégré à l'entreprise et qu'il se confonde parfois aux yeux des responsables informatiques avec le personnel interne ne suffit pas à le protéger. L'entreprise ne se sent pas les mêmes obligations et les mêmes devoirs vis-à-vis d'un personnel qui légalement n'est pas le sien ; s'il faut choisir, c'est l'analyste programmeur en régie, même présent depuis plusieurs années dans l'entreprise, qui partira et non pas son homologue interne.

L'apparente proximité des deux modes ne peut ainsi pas masquer des différences de traitement très claires : *"C'est quand même moins cher d'utiliser de la régie, parce que l'on joue sur les différences de convention collective. Quand vous appartenez au groupe, on provisionne en valeur actuelle la retraite qu'on va être amené à vous payer, etc."* (K) ou encore : *"La régie est le moyen d'avoir des effectifs présents mais non salariés. Le but est d'avoir une flexibilité pour le jour où les travaux sur lesquels vous travaillez vont finir. La régie permet d'avoir des personnes mais d'être flexible quand le niveau de développement, d'activité, de paramétrage baisse."* (F)

Si pendant les périodes de croissance a pu se développer l'illusion de la similarité des deux modes, interne et régie, la crise joue le rôle de révélateur : elle a fait apparaître la fragilité de ces contrats relationnels, même lorsque le lien est construit par des années de coopération ; elle a surtout provoqué le changement d'horizon temporel des contrats de régie.

Un autre facteur semble également avoir joué en défaveur des contrats de régie longue durée : dans quatre entreprises qui utilisaient beaucoup ce type de contrats, nos interlocuteurs ont souligné l'inadéquation de la régie longue durée au contexte technologique actuel.

2.2.2 Un mode déqualifié

Il semble que certaines entreprises n'ont pendant longtemps pas eu de grandes exigences de qualification pour le personnel en régie. Comme nous l'explique un responsable des études : *"Jusqu'à présent, l'appel à la sous-traitance était plutôt en régie, sous contrôle interne. On n'avait pas une forte exigence en matière de valeur ajoutée sous forme de méthodes ou de maîtrise technique. C'était des gens à l'image des analystes internes, venant en complément et pilotés par des chefs de projet internes. La plupart du temps, on était sur des interventions longue durée. Quelqu'un qui venait un an, deux ans, et donc du coup il développait une compétence globale sur le système à peu près au même niveau qu'un interne. C'était tout bonus pour les deux partenaires, parce que c'était des gens qui ne coûtaient pas cher à gérer pour le fournisseur, dont on privilégiait la capacité d'intégration, le relationnel et puis le professionnalisme dans le domaine. Mais il n'y avait pas de volonté d'apporter des compléments spécifiques techniques."* (H)

Plus qu'une compétence technique ou méthodologique, c'est la capacité d'intégration aux équipes internes qui est traditionnellement privilégiée : *"En régie, on a des gens qui ont un profil très polyvalent ou en tous cas interchangeables avec ceux de mon équipe. On ne recherche pas une technique particulière"*, (A4) ainsi que l'assimilation des spécificités du système d'information de l'entreprise : *"La régie, ça s'exerçait dans un cadre où on avait un système qui était extrêmement peu documenté, des chaînes existantes de douze ans d'ancienneté, extrêmement complexes, par strates."* (A4)

L'intégration très forte des équipes en régie longue durée dans l'entreprise, qui est leur avantage distinctif, pourrait se retourner au bout du compte contre elles.

Sélectionnés au départ pour leur connaissance des matériels et langages employés par l'entreprise, les salariés en régie sont orientés vers des tâches qui nécessitent un apprentissage très spécifique (prise de connaissance du système d'information de l'entreprise, "plongée" dans de vieux programmes peu ou mal documentés) ; leur niveau de connaissances générales

n'évolue pas, ce qui correspond à une déqualification progressive dans un domaine où la technologie se renouvelle rapidement. L'apprentissage des spécificités des systèmes existants n'a d'utilité que dans le cadre restreint de l'entreprise qui emploie le personnel en régie et ce uniquement tant qu'ils ne sont pas remplacés par de nouveaux systèmes.

A l'intérieur de l'entreprise, le personnel en régie se voit confié les tâches jugées moins "intéressantes" et moins qualifiées ; on lui attribue ainsi :

- la programmation plus que l'analyse : *"L'essentiel de l'analyse est faite par des gens en interne. Donc les gens en régie font beaucoup de programmation"* (A4), et, dans un autre groupe, : *"pour la réalisation, on fait appel soit à des externes, soit à des internes. Pour la partie analyse, on fait appel à des internes"* (C3),

- la maintenance plus que les nouveaux développements : *"Internes ou externes, il n'y a pas grande différence chez nous. La seule différence c'est que l'on privilégie autant que c'est possible les internes pour les mettre sur des projets nouveaux"* (C3),

- l'exécution plus que la conception des tâches : *"Jusqu'à ce jour on a essayé de ne pas sous-traiter d'études donc très peu de sous-traitance en analyse, ou alors c'est de la sous-traitance d'accompagnateur analyste. On intègre alors un extérieur à une équipe interne, sa tâche essentielle étant de gratter les tâches et quelqu'un de chez nous valide."*(E2)

Au total, le personnel en régie peut se trouver après un certain temps "déqualifié". Il est clair que ce n'est pas à l'entreprise d'assurer la qualification d'un personnel qui n'est pas le sien. Cette responsabilité incombe au fournisseur. Mais justement, du côté du fournisseur extérieur, il nous semble qu'ont parfois manqué des efforts sérieux pour compenser la déqualification des personnels placés en régie. Il faut dire que les incitations sont grandes à ne rien faire : à quoi bon extraire régulièrement l'informaticien en régie longue durée de son entreprise d'accueil (entreprise à qui cela posera en règle générale des problèmes de disponibilité et de répartition des charges), pour lui faire suivre une formation (et donc encourir des coûts supplémentaires et baisser *de facto* la rentabilité immédiate de la régie), formation qui a de toute façon de grandes chances de porter peu de fruits, étant donné que, de retour à l'entreprise, cet informaticien

retrouvera les vieilles chaînes écrites en Cobol et n'aura pas l'occasion d'utiliser ses nouvelles connaissances ?

De ce fait, certaines remarques que nous avons pu faire sur le désarroi et l'inexpérience des équipes internes face aux nouvelles technologies nous semblent également valables dans le cas des personnes sous contrat de régie longue durée. Nous pensons que ce peut être là une explication du résultat constaté lors de l'analyse empirique, à savoir qu'une forte nouveauté technologique est associée à une externalisation de type forfaitaire mais **pas** à une externalisation de type régie.

2.3 L'évolution vers un mode de fonctionnement plus proche du marché : la "régie forfaitée"

Le problème traditionnel de la régie est bien connu ; c'est celui de l'incertitude sur les coûts et du sentiment d'insécurité que cette incertitude génère : *"La régie, le management n'a aucune vision sur le temps que cela va durer. Il paye au fur et à mesure que les jours passent"* (F). Engagé dans un mode d'évaluation appuyé sur l'observation du comportement (l'équipe en régie travaille sur les lieux de l'entreprise, encadrée par des internes), le fournisseur extérieur ne s'engage pas sur le résultat de la prestation et ne donne pas d'estimation *a priori* de ce résultat.

De quels moyens dispose l'entreprise pour piloter les équipes en régie ?

Le premier moyen, qui a été largement utilisé et qui est à la base du système de régie de longue durée, est l'intégration. Le personnel en régie établit des liens personnalisés avec l'entreprise ; il fait siens les objectifs internes ; une relation de confiance se construit de part et d'autre. De cette façon peuvent être limités les comportements opportunistes.

Trois conditions nous semblent nécessaires à l'établissement d'un tel système. Tout d'abord la durée, qui permet l'intégration. Ensuite une certaine communauté d'intérêts entre le fournisseur et son client ; cette communauté d'intérêts est possible si l'on envisage la relation à long terme, chacun ayant avantage à satisfaire l'autre pour préserver le lien. Enfin, il faut que la

confiance puisse s'établir ; cette confiance requiert également une certaine pérennité de la relation.

Avec le déclin de la régie longue durée et la fragmentation de ce type de contrats, les possibilités d'intégration diminuent. **Les organisations ont mis en place d'autres moyens de contrôle, liés à un engagement sur le résultat** : c'est la "régie forfaitée", mise en place à des degrés divers par les différentes entreprises que nous avons étudiées, souvent sous la pression des contraintes budgétaires : *"La régie peut être une régie contrôlée. Il y a une partie qui est financée sur le budget informatique interne, et une partie sur des projets particuliers, appartenant au budget des utilisateurs. Sur la partie projet, comme on travaille avec un budget qui n'est pas le nôtre, si on obtient un engagement de la part de l'utilisateur de 80 jours, on demande à la personne en régie de s'engager sur 80 jours. On n'est pas encore au forfait, c'est de la régie contrôlée. S'il y a dépassement, on le prend à notre charge ; on ne répercute pas sur la SSII le forfait que nous font les utilisateurs."*(C3)

La "technique" consiste à fragmenter le plus possible l'engagement de la régie et à associer chaque engagement à une tâche précise, dont on peut estimer la durée : *"On constitue des lots et on fait s'engager la société conjointement avec nous sur un lot. Et à partir de là, elle travaille comme si elle était au forfait. Le mode juridique, c'est la régie ; mais les impératifs de résultat, les pénalités de retard, les obligations de déboguer sont comme s'ils étaient au forfait"*. (D)

Ce responsable informatique nous vante les mérites de ce système : *"Dans ma formule, si le prestataire est en avance, on le voit bien puisqu'il travaille ici, donc on comble par autre chose, ce n'est pas de la marge pour lui. S'il est en dépassement, c'est à lui de combler puisqu'il s'est engagé au départ sur le nombre de jours pour le lot. On a un cas maintenant où il y a des gens qui vont faire 10 à 15 jours gratuitement pour combler le déficit. L'intérêt du prestataire, c'est de garder le client et de bien expertiser le lot dès le départ de manière à avoir un débat contradictoire et se mettre d'accord sur le nombre de jours. Mais une fois qu'il*

s'est engagé, il le fait. On a le contrôle de pilotage du projet, mais il est libre de proposer toute modification de nature à améliorer son travail." (D)

De fait, on en revient à une solution très proche du mode forfaitaire, avec les obligations de celui-ci, une pression due à la juxtaposition de "micro-contrats" mais également toutes les exigences liées à l'observation directe du comportement par le client. Ce type de marché est contraignant pour les sociétés de services. Il est également contraignant pour l'organisation interne, car il l'oblige à maintenir des systèmes coûteux d'encadrement des tâches et d'observation du comportement. Il permet en revanche de gérer des situations où une externalisation au forfait s'avère impossible :

"Si on est très clair sur un cahier des charges, on fait du forfait. J'aurais presque tendance à vous dire on fait de la régie quand on n'est pas clair."(F)

"Ceux qui sont en régie sont sur des prestations que je ne sais pas forfaiter, par exemple dans le cas d'applications où on les prend en amont du cahier des charges, ou sur des petites opérations de programmation, c'est sur des choses pour lesquelles je n'ai pas un cadre assez fixe. Mais j'essaie de forfaiter au maximum."(A2)

Les entreprises que nous avons étudiées entendent en règle générale restreindre globalement l'utilisation de la régie. Pour être plus précis, le forfait est apparu dans le discours de nos interlocuteurs dans 65 % des cas comme le "mode désiré", la solution d'avenir, l'évolution logique.

III. L'externalisation au forfait : un mode souhaité mais des problèmes de contrôle

Comme nous l'avons indiqué, une majorité de nos interlocuteurs expriment un **enthousiasme de principe pour le forfait**. Cependant dans la pratique, ce mode paraît encore peu utilisé par les entreprises que nous avons consultées¹². Notre objectif est ici de tenter de

¹². A l'exception d'un cas particulier sur lequel nous revenons au § 3.3 de ce chapitre.

comprendre et d'expliquer les raisons de cet écart entre le désir exprimé et la réalisation concrète.

Nous nous intéressons dans un premier temps aux situations dans lesquelles les entreprises déclarent utiliser le forfait, puis aux motifs qui les font renoncer dans d'autres cas à ce mode contractuel. Nous tentons enfin de définir ce que pourrait être un "bon usage" du mode forfaitaire et les apports de celui-ci pour l'entreprise, y compris sur l'amélioration des modes de fonctionnement interne.

3.1 Les situations d'utilisation du forfait

Ainsi que nous avons pu le montrer dans l'étude quantitative, le forfait paraît associé à la recherche d'une **compétence technologique** pointue :

"Ce que l'on cherche dans un cadre forfaitaire, ce sont des compétences pointues, fines, de spécialistes que l'on n'a pas forcément dans l'équipe."(A4)

"On a peu forfaits actuellement, on commence à en faire sur des développements systèmes, des outils de communication PC/Vax, assez spécifiques."(H)

De même, la structure du projet était-elle apparue dans l'étude quantitative comme corrélée positivement au choix du mode forfaitaire. Nous avons choisi cette variable comme indicateur de la **possibilité de définir et d'évaluer le résultat a priori** ; le discours de nos interlocuteurs confirme l'importance de ce second point :

"Des forfaits, on en fait dans quelques cas relativement simples où on sent que le besoin est très bien cadré, très stable, et techniquement, on y voit très clair."(K)

"Le forfait, il faut être capable d'exprimer ce que l'on souhaite. De toute manière, il ne faut pas s'être trompé sur les spécifications, faute de quoi on prend le coût des corrections. La difficulté est de définir le besoin."(F)

"Le forfait, c'est une tendance. Il y a une forte demande des SSII. Une forte demande interne aussi, théorique en tous cas, politique, pour développer des forfaits. En théorie c'est très intéressant. En pratique, ce n'est réalisable que sur des sujets bien maîtrisés, bien

identifiés. (...) Les forfaits on les fera beaucoup plus sur des opérations nouvelles, et ils seront d'autant meilleurs que le sujet sera bien délimité."(B)

Enfin, le choix d'un progiciel peut dans certains cas engager l'entreprise à retenir un forfait global : *"Des forfaits, on en fait avec des fournisseurs de progiciels, pour des adaptations."*(H) La justification est double : le fournisseur maîtrise mieux l'environnement technique du progiciel (environnement pas forcément plus complexe sur le plan des technologies employées mais inconnu pour les équipes internes) ; le progiciel permet d'avoir une base stable pour l'évaluation du besoin.

Plus instructive encore est l'étude des situations et des motivations qui conduisent à écarter le mode forfaitaire. Certains de nos interlocuteurs ont été particulièrement éloquents sur ce point que nous allons maintenant aborder.

3.2 Les freins au développement du forfait

A l'analyse de l'ensemble des entretiens, nous avons identifié deux freins à l'utilisation du forfait. Le premier semble être la difficulté à définir précisément le résultat attendu. Le second, qu'il faut à notre avis analyser comme une conséquence du premier, est la crainte d'une relation difficile avec le fournisseur, engendrant des coûts de transaction élevés.

3.2.1 La difficulté à définir le résultat attendu

La réalisation d'un cahier des charges complet, décrivant de façon exhaustive la prestation, pouvant servir de base à un contrat et permettant d'évaluer *a posteriori* le fournisseur semble être l'un des principaux freins à la généralisation des forfaits :

"On a fait de la régie parce qu'on ne pouvait pas faire un cahier des charges ; c'était presque impossible d'estimer au départ la charge." (A2)

"C'est difficile de définir ce que l'on veut pour faire des forfaits. Je préfère naturellement les contrats forfaitaires, mais seulement s'il existe une description des tâches très très précise, et c'est le problème souvent". (G)

L'intégration de la nouvelle application au système d'information, qui était apparue comme un facteur défavorable au fournisseur sur le critère industriel, ressort également ici comme un handicap à la définition du forfait : *"Sur des applicatifs comme les nôtres, qui ont des milliers de programmes assez interconnectés et où il faut une très forte connaissance de l'existant, c'est assez difficile de brancher quelqu'un de l'extérieur... On considère qu'il faut 6 mois pour former quelqu'un de compétent, alors faire un forfait..."(B)*

L'évaluation du coût de la prestation n'est pas plus simple : *"Le problème de fond, c'est que l'on ne sait pas évaluer. Et ne sachant pas évaluer, nous avons des sous-traitants qui ne veulent pas prendre de risque et nous, on ne veut pas non plus en prendre. On prend un risque en interne mais au moins on n'est pas en face de sous-traitants qui prennent des marges larges pour éviter le risque de baisser la culotte. C'est très difficile, on n'a jamais su évaluer en informatique. On croit que l'on sait, on ne sait pas."(L)*

Le fournisseur pourrait effectivement être tenté de sur-évaluer au départ sa prestation pour absorber une partie de son risque. Mais même sans juger *a priori* de l'attitude du fournisseur, le cahier des charges peut s'avérer insuffisant pour lui faire comprendre la totalité du besoin :

"On hésite à faire du forfait. Il y a eu dans le passé des expériences malheureuses et la grande leçon, c'est que l'on n'est pas sûr que l'on parle de la même chose quand on fait du forfait. On a des méthodes de développement au niveau du groupe très particulières, il n'est pas question de ne pas les utiliser. La SSII "x" en a d'autres qui sont peut-être aussi valables mais si on lui demande d'en changer, et qu'elle n'a pas tout à fait conscience de la conséquence sur ses coûts de prestation, et bien on va vers un clash au niveau du forfait et on se retrouve en perte. Autre exemple, on est vraiment très stricts en matière de sécurité informatique, beaucoup plus que la moyenne du marché français, ce qui se retrouve évidemment au niveau de la programmation, droits d'accès, ce dont la SSII n'est pas non plus forcément consciente. Je vais lui dire attention, on est très strict sur ces sujets là, donc le gars

va prendre 10% de plus. Tant qu'il n'a pas touché le fond véritablement du problème, il n'est pas capable de faire un forfait raisonnable. D'où l'idée de dire, on travaille en régie."(K)

3.2.2 La perception d'un risque élevé

Tous les interlocuteurs ont la perception d'un risque assez élevé en situation de forfait. Beaucoup nous ont raconté ce que l'un d'entre eux a appelé "les histoires d'horreurs du mode forfaitaire". Inadéquation des résultats, manque de contrôle, rigidité contractuelle, opportunisme des fournisseurs¹³, peuplent ces histoires vraies.

Inadéquation des résultats :

"Ce qui s'est passé pour la mise en place de MSA GL, c'était sous la responsabilité de la société X, qui a développé des états usine à gaz, qui est arrivée avec je ne sais combien de personnes, il y en a eu pour je ne sais combien de millions, en tout cas c'est monstrueux. Tout ce qui a été développé a pratiquement été mis à la poubelle."(C1)

Manque de contrôle :

"Le sous-traitant nous a dit j'ai des bêtes à programmer, j'ai une démarche industrielle et ça n'a pas marché. C'était une société de services dirigée par un grand prêtre de la méthodologie de développement, qui est connu dans le monde informatique et qui nous a dit : ne me dites pas ce qu'il y a dedans, je me fiche du sujet, je fais cela en deux fois moins de temps que les autres. Et il s'est planté complètement." (L)

"Les forfaits gigantesques, on débouche souvent sur des procès ou des conflits (...) Je n'ai jamais vu de grand projet au forfait dans lequel l'on savait à l'avance où l'on allait." (B)

Lourdeur et rigidité contractuelle :

"On a fait le cahier des charges en prenant une société en régie parce qu'on n'avait pas les équipes pour le faire compte tenu des délais et ensuite on a continué en régie, parce que le

¹³ . Tous ces avatars n'étant en fait que des variétés différentes de coûts de transaction !

fait de forfaitiser, on aurait perdu ne serait-ce que le temps passé pour faire les contrats, tout le reste... ." (A2)

"Le projet a dérapé parce que l'utilisateur n'était pas stabilisé et le fournisseur a dit je m'en fous, je continue." (L)

"Sur un développement vous avez toujours l'utilisateur qui revient pour vous dire qu'il s'est trompé, même s'il a signé en bas de toutes les pages de spécifications. Et il refuse la livraison d'une application qui n'est pas up to date. C'est l'éternel problème. Quand vous avez un forfait, vous avez un prestataire de services qui vous dit moi je m'en fiche, je suis au forfait et je respecte mon forfait. Et c'est la quadrature du cercle parce que vous avez d'un côté un fournisseur qui a raison et de l'autre un utilisateur qui a raison. Dans ma vie professionnelle, j'ai toujours vu des forfaits malheureux." (L)

Mais c'est sur le thème de l'opportunisme des fournisseurs que quelques interlocuteurs ont développé le plus longuement leur argumentation :

"Il y a une exaspération totale vis-à-vis de l'informatique, de l'attitude des constructeurs et des prestataires de service même dans les très grands projets. L'attitude qu'ils avaient dans les années d'euphorie, c'était de promettre dans 3 mois pour 3F ce qu'ils arrivaient à faire en 3 ans pour 18F, en expliquant aux gens qu'ils n'y connaissaient rien, qu'ils avaient mal formulé leurs besoins, etc... Vous vous faites avoir une fois. La deuxième fois, vous faites cahier des charges, contrat, etc... Cela ne change rien, les gens disent on a signé mais ce n'est pas grave, vous allez quand même payer le double, je ne termine pas, je ne tiens pas le forfait. Vous payez. Ou vous allez au contentieux. On est en train de gérer un projet comme cela, à la barre de fer, contre une SSII : pré contentieux permanent. Plis recommandés, échange de courriers, avocats..." (J)

Le sentiment est parfois celui d'une frustration vis-à-vis d'un partenaire qui ne joue pas le jeu :

"On a l'habitude dans notre métier de faire des travaux. Vous allez voir un maçon, je veux un mur comme ça, ça va vous coûter 10, ce sera fait le tant. A la sortie, il dit cela ne m'a pas

coûté 10, cela m'a coûté 11. Pourquoi ? Parce que... S'il s'est trompé dans ses calculs, s'il a dit que cela faisait 10m et qu'il y en avait 11, c'est son problème, ce n'est pas le mien! Or les SSII me disent tout ce que l'on découvre après, c'est votre problème. Alors qu'ils nous disent au départ, on a très bien compris votre problème, on a des outils merveilleux, on s'engage sur des forfaits. Et après, au bout d'un an de dépassement... " (J)

"Le fournisseur nous a dit qu'il utilisait les mêmes langages et outils que nous, mais en fait c'était faux." (L)

"Quand on est au forfait, il y a deux cas de figure, soit la personne réussit à faire moins que ce qui était prévu, et au quel cas, c'est de la marge pour lui, soit il fait plus et il s'arrange toujours pour vous faire comprendre que c'est de la faute du client et il y a des avenants. De toute façon on se fait toujours avoir". (D)

3.3 Le "bon usage" du forfait : les conditions de mise en oeuvre

Comment éviter ces "histoires d'horreur" ? Pour l'ensemble de nos interlocuteurs, la clé de la solution se trouve dans la **qualité de la définition, du suivi, et de l'évaluation du résultat** produit par le fournisseur¹⁴.

Dans l'ensemble des entreprises étudiées, des réflexions sont en cours pour définir un mode de gestion approprié des contrats au forfait. Ces réflexions sont plus ou moins avancées. Une entreprise de notre échantillon se distingue sur ce point. Dans cette entreprise, une **politique de sous-traitance au forfait** a été mise en place concernant un domaine d'activité (A3) ; elle intègre une démarche complète visant à garantir les conditions de mise en oeuvre de ces contrats. Nous détaillons ici cet exemple qui nous semble riche d'enseignements.

3.3.1 Une analyse des besoins en interne, avec des outils adaptés (prototypage)

Premièrement, tout projet que l'entreprise envisage de sous-traiter au forfait démarre par une étape interne d'identification des besoins très complète, dans laquelle on utilise souvent des

¹⁴. On ne s'étonnera pas de cette constatation, tout-à-fait conforme aux hypothèses dérivées des théories de l'agence et des coûts de transaction.

techniques de prototypage afin de préciser le contenu. L'objectif de cette étape est la définition du résultat *a priori*, qui doit être la meilleure possible pour limiter les ajustements ultérieurs et par voie de conséquences réduire les occasions de conflit avec le fournisseur :

"Je crois que plus nous mettons de rigueur, plus nous faisons des spécifications précises et moins il y a de dérive possible. Il y a des sujets que l'on domine bien et en général cela se passe bien, parce que on sait spécifier ce que l'on veut. Et puis il y a des projets plus novateurs, où l'on n'a pas d'expérience. Ce que je préfère faire quand on ne sait pas très bien, c'est passer d'abord chez nous par beaucoup d'étapes de maquettage. On défriche un peu le sujet de notre côté pour y voir clair, pour bien voir les risques que l'on prend après sur les développements. Donc on maquette, on mesure tous ces risques. On essaie d'apprendre. Puis après seulement on spécifie. On essaie d'être très précis sur les spécifications fonctionnelles et sur l'ergonomie."

Le **prototypage** (la maquette) s'avère un **moyen privilégié** de discussion avec les utilisateurs: *"Les spécifications sont systématiquement envoyées aux utilisateurs pour qu'ils les valident. La maquette est très importante parce que autant ils vont s'intéresser à une maquette, ils vont bien vouloir la valider, autant lire 200 pages, personne ne les lit. Et après évidemment, ce n'est pas de leur faute. Le moyen maquette est cent fois supérieur, parce qu'ils acceptent volontiers de passer une journée sur une maquette, de jouer un peu avec."*

(...) "Les maquettes vont assez loin. On travaille avec l'utilisateur pour le faire réagir sur les écrans, sur les enchaînements. Ça fait partie du cahier des charges. J'ai fait des appels d'offres en donnant une disquette et en disant, on veut cette ergonomie. Avec des outils de maquettage. Ce sont des maquettes, c'est à dire des choses perdues, mais qui ont permis de régler tout cet aspect interface homme / machine qui est vraiment très important. C'est souvent là-dessus que l'on bute, parce que l'utilisateur n'a pas imaginé des cas de figure et nous, on n'a pas pu les imaginer."

Cette analyse des besoins très fouillée, dont le produit fini est une maquette ou un prototype, vise à compenser la relative rigidité des phases suivantes : pour que le contrat se déroule bien, mieux vaut en effet éviter le plus possible les ajustements *a posteriori*. Dans cet

esprit, notre interlocuteur considère qu'une des plus grandes erreurs dans la conduite d'un projet au forfait serait de lancer trop tôt la réalisation, avant que la définition du projet ait eu le temps de mûrir en interne. De la même façon, il conseille d'éviter les interactions directes entre les utilisateurs et le sous-traitant : *"En général, les utilisateurs n'ont pas de relations avec les sous-traitants pendant la réalisation, pas du tout. C'est rare. Cela permet de se limiter aux spécifications de départ."*

3.3.2 Des plans qualité pour le suivi de la prestation

Le second point prend en compte tous les aspects de suivi de la qualité des prestations. L'utilisation de méthodologies, la définition et l'utilisation de plans qualité sont au coeur de ce sujet :

"Je crois que la sous-traitance demande de la rigueur des deux côtés. De notre côté dans les spécifications et dans le suivi des projets; du côté du sous-traitant cela demande de la rigueur et de l'organisation d'abord pour se mettre d'accord sur les techniques et sur la méthodologie que l'on va utiliser et sur la qualité de ce qu'il va apporter au développement. La sous-traitance est un très bon moyen si on l'encadre par une batterie de bornes, certifications qualité, plans qualité etc."

(...) "J'utilise de plus en plus des plans qualité. Je suis quelques développements avec des sous-traitants d'une façon assez rigoureuse et je trouve que cela marche très bien à condition que l'on ait cette rigueur là et également à condition que le sous-traitant lui aussi soit rompu à ce genre de choses. Avec une entreprise bien structurée et rompue à ces techniques cela va très bien marcher mais une entreprise qui n'est pas rompue à ce genre de techniques, c'est une catastrophe, c'est encore pire que si on ne faisait rien."

A la base du contrat de qualité, on trouve deux documents, signés par le sous-traitant. :

- le "plan de développement logiciel" : *"c'est un document qui détaille comment on va faire, cela explique qui fait quoi, les phases, le découpage en lots, etc."*,

- le "plan qualité" : *"le plan d'assurance qualité donne toutes les phases de contrôle et toutes les métriques que l'on va faire sur les développements. Les tests, les exigences qualité à chaque étape."*

Et, prenant l'exemple d'un contrat récent :

"Ce projet est un projet de développement de 4MF. On a des réunions de suivi mensuelles; le projet est sur 14 mois (début de développement à fin de mise en place des sites pilotes). On a des phases de validation d'étape avec des réunions et des documents de validation pendant toute la phase conception / développement. On a fait des tests, un peu de métrique sur certains lots déjà développés pour voir si les sous-projets vont dans la bonne direction pour atteindre les objectifs. Par exemple, pour un sous projet de communication entre un client et un serveur, on a fait une validation avec des tests de performance. Les étapes sont parfaitement décrites dans le plan qualité. Il y a un ensemble de tests spécifiés, des résultats de tests et des choix qui sont faits derrière ces résultats de tests."

De ces documents composant le plan qualité notre interlocuteur nous dit : *"Ce sont d'excellents cadres; quand ils sont bien élaborés par les deux parties et quand on est bien d'accord sur leur contenu, après je trouve que c'est un confort formidable. Chacun sait ce qu'il doit faire, quand il doit le faire, les étapes... on n'a plus qu'à suivre."* Tout autant qu'un cadre pour l'organisation de la production, le plan qualité est un cadre pour les ajustements intervenant en cours de projet ; il garantit de plus une certaine visibilité du client sur les travaux réalisés par son fournisseur.

3.3.3 Une phase de tests amplifiée

Troisième élément, la phase d'évaluation du résultat (contrôle et tests) prend une importance considérable ; elle est pré-définie dans le plan qualité :

"A l'automne on va commencer les phases de tests après intégration, chez le sous-traitant d'abord ; nous lui avons fourni les données de tests, il nous donne les résultats. Ensuite il y a une deuxième phase de tests chez nous ; nous testons les logiciels théoriquement terminés."

Troisième phase de tests sur site pilote ; là on fait faire les tests par les utilisateurs. En tout, c'est un mois de tests chez le sous-traitant, un mois de tests ici, deux mois en site pilote chez l'utilisateur. Sur un projet de 14 mois, nous avons 4 mois de tests et validation. C'est un tiers de conception, un tiers de développement et un tiers de tests."

3.3.4 Des règles d'ajustement strictes sur le principe, souples dans l'application

Enfin, les règles d'ajustement avec le fournisseur sont un mélange de sévérité annoncée, conforme aux règles du marché, et d'ajustement coopératif, ce afin de préserver le sentiment d'équité et le rapport de confiance entre les deux partenaires : *"Quand il y a des ajustements, on négocie. Hier, il y a eu un petit problème d'ajustement sur des points d'ergonomie. Après le maquetage, des utilisateurs ont préféré que ce ne soit pas tout à fait comme c'était spécifié. On a conclu sur quelques jours d'ajustement. Une partie à la charge du sous-traitant et une partie à notre charge, parce qu'il faut trouver un compromis à tout cela."*

Cette expérience est confirmée par d'autres entreprises de notre échantillon pratiquant également le forfait :

"La règle du jeu est a priori pas d'ajustements, mais on est quand même dans une relation avec un partenaire. Si vraiment sur un truc il s'est planté, il va venir nous voir et nous dire, on revoit le devis, on n'est pas..." (I2)

"Si c'est un forfait, moi ce qui m'intéresse, c'est le résultat du forfait en termes de délai et de coût. A l'intérieur... Mais la plupart des sociétés jouent la transparence. Cela leur permet de demander une petite rallonge s'ils se sont trompés ; on l'accorde généralement parce qu'on sait très bien que si on ne l'accorde pas alors que l'on s'est trompé l'un et l'autre dans l'évaluation, c'est le projet lui-même qui va en pâtir." (E2)

3.3.5 Des recommandations complémentaires

Des règles supplémentaires peuvent être appliquées pour faciliter la mise en oeuvre des forfaits ; un de nos interlocuteurs cite ainsi la taille des contrats : *"Plus le champ est limité,*

mieux c'est. Il est facile de développer une petite application micro, sans connexion avec l'existant, c'est un forfait typique facile à réaliser." (B)

Deux autres choisissent d'inclure un membre de l'équipe interne à l'équipe du fournisseur qui réalise le forfait, ce afin de faciliter la reprise en main *a posteriori* : *"J'essaie d'adjoindre une personne dans l'équipe de sous-traitance. Si on veut garder la maîtrise de application, en faire la maintenance ou dans le cadre d'un progiciel faire la maintenance du paramétrage, on met des personnes de l'équipe" (A4)* et : *"Lorsqu'il y a des réalisations au forfait, dans la mesure du possible, j'inclus quelqu'un du groupe au projet. Cette personne va participer ensuite à la recette et au lancement, assurer la maintenance ou au moins garantir la maintenance." (E2)*

Enfin, les fournisseurs proposent également des solutions afin de mettre en confiance leurs clients et de contrecarrer les présomptions d'opportunisme dont ils font l'objet. On retrouve ici une forme de "coûts d'obligation", définis par la théorie de l'agence, et qui correspondent à l'émission de signaux attestant de la qualité de la prestation. La proposition d'un plan qualité par le fournisseur répond à cet objectif, tout en palliant une déficience éventuelle chez le client :

"Quand on fait des forfaits, c'est le contrat du fournisseur plus son plan qualité qui sert de base à toute la discussion. Comme on n'est pas très mûrs sur les plans qualité on prend ceux des fournisseurs. On n'est pas en mesure d'imposer les nôtres." (L)

Gérer un forfait demande une rigueur supplémentaire de la part de l'organisation ; **la mise en oeuvre de forfaits peut ainsi également être l'occasion d'améliorer les modes de fonctionnement internes.**

3.4 Le forfait comme mode d'apprentissage interne

"On veut faire du forfait parce que c'est un levier pour plus de rigueur. C'est essentiellement cela que je vois comme intérêt."

(...) "Je ne sais pas si le forfait est vraiment moins cher, je n'ai pas d'idée. Mais avec le forfait, il y a deux effets de levier. Tout d'abord, c'est plus "challenging" pour le prestataire,

c'est un premier levier. Et puis il y a un deuxième levier sur l'interne, qui est l'approche avec des méthodes un peu structurées. Parce que l'on a tendance avec une organisation très réactive comme la nôtre à travailler un peu comme des "sagouins". Depuis quatre / cinq ans, on a toujours eu une forte volonté de structurer, mais il y a toujours eu le contrepoids de l'urgence. Le forfait peut être un élément qui contribue à nous imposer à nous mêmes de rédiger des cahiers des charges un peu précis." (H)

La plupart de nos interlocuteurs, sans avoir atteint le degré d'avancement de l'entreprise dont nous venons de décrire l'expérience, souhaitent pratiquer plus régulièrement le forfait et commencent à définir des méthodes et des modes de fonctionnement adaptés à ces conventions. Ils découvrent alors les **bénéfices indirects** induits par la mise en oeuvre de ce type de contrats sur l'organisation interne.

La régie n'avait pas modifié les pratiques internes ; sous sa forme traditionnelle, elle n'a pas inféré d'apprentissage sur la conduite des projets. Elle n'a pas non plus joué véritablement le rôle de pression concurrentielle ; comme nous l'explique un responsable d'applications ayant remplacé récemment dans le cadre d'un contrat de maintenance une ancienne organisation en régie par un forfait : *"D'habitude on avait des gens en régie. Pour les informaticiens internes, cela ne menace en rien leur compétence, leur maîtrise, etc." (A4)*

Ce que la régie n'a pas provoqué, le forfait semble pouvoir le faire. Nous retrouvons ici des conclusions proches des hypothèses formulées par Quinn, Doorley et Paquette (1990, cf. chapitre 4) sur les bénéfices indirects de l'externalisation. Il convient toutefois de souligner encore une fois que **toute forme d'externalisation ne permet pas de générer ces bénéfices indirects** ; nos résultats montrent que la régie n'a pas eu cet effet. Pour que l'externalisation soit une source de bénéfices indirects, il faut que plusieurs conditions soient remplies, ce que certains modes de forfait semblent parvenir à accomplir.

3.4.1 Les bénéfices indirects induits par le forfait

Nos interlocuteurs voient deux principaux avantages indirects de la mise en oeuvre de contrats forfaitaires, jouant sur l'organisation interne. Le premier concerne les **méthodes, outils et modes d'organisation**, qui évoluent dans le sens d'une plus grande rigueur ; le second porte sur la **motivation** du personnel interne.

Nous l'avons vu, les contrats au forfait peuvent donner lieu à des coûts de transaction très élevés s'ils ne sont pas encadrés par une démarche particulièrement rigoureuse. Plus précisément, ce sont les ajustements *a posteriori* qui risquent de générer ces coûts de transaction, l'entreprise étant alors engagée dans une situation de "petit nombre" où elle est liée à son fournisseur, ce d'autant plus que les travaux sont avancés (cf. chapitre 6). Afin d'éviter ce risque, il faut mettre en place des garanties, tant au niveau de la définition des besoins *a priori* que du suivi de la qualité de la prestation ou du contrôle et de l'évaluation des résultats. La mise en place de ces garanties représente bien évidemment - comme tout système de contrôle - un coût supplémentaire ; dans le cas de la relation avec un fournisseur extérieur ce coût supplémentaire est justifié par la diminution des risques.

L'ensemble de ces méthodes et outils associés à la pratique des contrats au forfait est souvent mis en place au début avec l'aide de fournisseurs extérieurs qui les proposent comme partie intégrante de leur prestation ; l'entreprise peut alors "apprendre" grâce à son fournisseur extérieur.

Considérons par exemple le cas de ce contrat de maintenance au forfait qui a remplacé dans l'une des entreprises une ancienne équipe mixte (interne et régie) et qui nous semble caractéristique de la situation d'apprentissage : *"La tierce maintenance marche même mieux que nos équipes internes. Ce sont des gens très méthodiques. Ils ne peuvent pas se permettre de bâcler une recette. Les miens, quand ils n'ont pas le temps, la documentation n'est pas faite. Eux, c'est inscrit dans le contrat de service, je peux à tout moment voir la documentation. Donc j'ai une qualité qui est supérieure à celle que j'avais en interne. Ils sont un peu plus coûteux, parce qu'ils documentent et testent de façon très systématique. Cela*

parce que derrière, s'il y a le moindre problème au niveau de la mise en exploitation, ils sont obligés de refaire gratuitement. Donc ils ne font strictement aucun cadeau. Quelques fois, ils vont être contraignants pour nous, demander des délais plus longs qu'en interne. D'accord, mais ils testent, ils documentent. Globalement ils coûtent un peu plus cher que mon équipe, mais en termes de qualité, j'ai un gain très appréciable." (A4)

L'entreprise a souhaité utiliser ce forfait comme mode d'apprentissage pour ses équipes internes : *"On avait le choix. Soit des gens qui travaillent à distance, soit des gens qui travaillent ici. Je pensais que c'était intéressant d'avoir les gens qui avaient une approche méthodologique importante au sein de nos équipes de maintenance pour que les gens puissent voir de façon concrète comment ils travaillent et la façon dont nous on pouvait reprendre ce genre de techniques"*. Même si le forfait est effectué à distance, nous pensons que l'entreprise cliente peut expérimenter et apprendre avec son fournisseur la conduite d'une relation formelle appuyée sur des documents de type "plan qualité", des procédures de suivi et la définition de métriques d'évaluation.

L'étape suivante a été pour cette entreprise de mettre en place les mêmes plans qualité (contrats de service) en interne que ceux utilisés pour gérer les fournisseurs extérieurs : *"Avec l'équipe de tierce maintenance au forfait, on a un contrat de service. On a une garantie sur la modification qui fait que s'il y a un incident après mise en exploitation, on va le suivre dans un contrat qualité. En interne, cela commence. On a quelques contrats de service qui ont été signés avec certains de nos clients internes, cela va devenir la tendance. Parce qu'il faut que l'on puisse se donner les moyens de garantir que ce que l'on fait est de bonne qualité." (A4)*

La relation contractuelle au forfait induit également une plus grande **rigueur dans l'étape d'analyse du besoin**. Nous avons pu constater dans l'étude quantitative que, sur les dix-neuf projets ayant fait l'objet d'un forfait, treize avaient donné lieu à la réalisation d'une maquette ou d'un prototype, ce qui semble considérable. Dans deux entreprises, les techniques d'analyse développées dans ce cadre ont essaimé en interne. Le prototypage est d'abord utilisé pour les

projets au forfait afin d'avoir un cadre contractuel mieux défini ; nous pensons qu'il peut devenir ensuite une méthode de travail utilisée également pour les projets internes.

Outre la rigueur, le second avantage est la motivation du personnel de l'entreprise ; celle-ci dérive de la mise en concurrence : *"Avoir un fournisseur extérieur, cela me permet d'avoir un levier sur l'informatique interne. Ça les met en concurrence, ça les motive, ils sont obligés d'avancer plus vite parce que les autres avancent plus vite."* (A4)

3.4.2 Les conditions pour que l'externalisation génère un apprentissage interne

Il reste que pour que les bénéfices indirects soient atteints, l'entreprise doit faire attention aux conditions de mise en oeuvre du forfait. A observer les situations et à écouter nos interlocuteurs, deux conditions principales semblent devoir être réunies.

L'analyse des dysfonctionnements internes faite dans le §1.3 de ce même chapitre (analyse d'une situation d'auto-contrôle ayant dérivé en "non-contrôle") permet de mettre en évidence la première condition. L'entreprise qui y est décrite utilise les contrats au forfait et les informaticiens internes ont même une bonne maîtrise des méthodes et outils de contrôle que nous avons exposés. Cependant le fonctionnement interne semble médiocre ; la rigueur utilisée pour gérer les fournisseurs extérieurs n'est pas appliquée, loin s'en faut, aux productions internes. Notre interprétation est que la raison principale de cet échec est la quasi-absence de contrôle des informaticiens internes par les utilisateurs. **Autrement dit, pour que l'externalisation puisse générer un apprentissage interne, nous pensons qu'il est nécessaire qu'il y ait en même temps une pression, exercée par la direction générale et les utilisateurs, qui aille dans le sens d'une comparaison des performances internes et externes et d'une "mise à niveau" des prestations du service informatique.** Encore faut-il pour cela que le forfait ne soit pas décidé uniquement par les informaticiens et que les utilisateurs aient une visibilité correcte sur les processus internes.

La seconde condition est presque inverse : il s'agit ici de **limiter l'effet de stress** associé à la mise en oeuvre de forfaits et de préserver le moral des équipes internes. La mise en oeuvre de forfaits peut en effet être perçue comme le premier pas vers le désinvestissement : *"Ils se sont dit on va nous remplacer par des gens. On est remplaçables. Et si l'on s'amuse à généraliser ce genre de politique, on n'a plus besoin d'informaticiens internes."* (A4)

Le récit qui nous a été fait de la mise en place de ce forfait de maintenance illustre bien ce point. Le forfait a été choisi parce que l'entreprise ne disposait pas des effectifs et des compétences en interne ; malgré cela, la décision a été perçue comme une menace par l'ensemble des informaticiens d'études : *"C'était quelque chose qui était extrêmement difficile à gérer en interne sur le plan de la relation. C'est huit mois où il y a eu des crises, c'était assez conflictuel. Jusqu'au moment où ils ont compris qu'effectivement, il n'y avait pas de menace. On a fait beaucoup de communication, non seulement au niveau de l'équipe mais aussi du reste pour qu'ils comprennent pourquoi on avait fait cela. C'était extrêmement perturbant. C'était un choix qui n'était pas neutre. Ce n'était pas je prends deux ou trois personnes de plus dans mon équipe, c'était complètement différent. C'était un choix politique. Ce n'est pas du tout la même chose que ce l'on faisait précédemment."* (A4)

Pour beaucoup d'informaticiens internes, la pratique de vrais forfaits (et non pas de "régie forfaitée") est encore une rupture par rapport aux habitudes antérieures. Elle peut être vécue comme une menace ; à charge de l'encadrement de faire comprendre qu'elle peut être un complément et un enrichissement -et de respecter ce contrat- si l'entreprise veut bénéficier des bénéfices indirects que les forfaits peuvent apporter à l'organisation interne.

III. Conclusion sur le problème du contrôle

Au terme de cette section, **la question du contrôle s'affirme bien à la fois comme une préoccupation majeure et comme une forte contrainte quant aux choix d'externalisation.**

De fait, le problème du contrôle se pose pour l'ensemble des trois modes d'organisation étudiés : informatique interne, externalisation en régie et externalisation au forfait.

Nous avons observé et décrit **trois modes distincts de pilotage interne**.

Le premier, que nous avons appelé "la sévérité du marché", est centré sur le respect des coûts et consiste à imposer à l'informatique interne une **logique de marché** (engagement sur le prix de la prestation associé à une contrainte forte sur le respect du prix). Les résultats obtenus paraissent efficaces sur le plan de la limitation des coûts mais la contrainte budgétaire semble avoir des conséquences tant sur l'organisation de l'activité que sur le choix des projets de développement.

Le deuxième mode de pilotage ("l'ajustement coopératif"), s'appuie sur la **coopération informaticiens / utilisateurs** et sur l'établissement de rapports de confiance. Pour être viable, il nécessite une transparence sur les modes de fonctionnement de l'informatique interne et le développement d'orientations managériales chez les informaticiens. C'est le mode de pilotage le plus favorable à l'établissement d'une synergie entre l'informatique et le reste de l'entreprise.

Le troisième mode de pilotage correspond à la situation **d'auto-contrôle** de l'informatique interne. Nous pensons que certains risques peuvent être associés à cette situation ; nous avons ainsi détaillé et commenté l'exemple d'une entreprise de notre échantillon, dans laquelle l'auto-contrôle a dérivé en situation de non contrôle.

Plus largement, il semble que l'on puisse établir un lien entre le type de contrôle interne et les choix d'externalisation. Les différences portent principalement sur deux éléments : qui décide de l'externalisation et quelle est la logique globale des choix :

Dans le premier cas, l'externalisation est décidée par la direction générale et les utilisateurs sur certaines tâches ou projets ; elle répond à une logique de coût et permet d'affirmer la sévérité des sanctions en cas de non respect du budget par les effectifs internes.

Dans le second cas, l'externalisation est décidée en commun par la direction générale, les utilisateurs et les informaticiens ; la comparaison avec un fournisseur extérieur sur le critère du

coût et surtout de la qualité à long terme des prestations est un moyen de légitimer l'informatique interne ; elle est pour les utilisateurs une garantie.

Dans le troisième cas, l'externalisation est décidée par l'informatique seule sur des critères opérationnels. Dans ce cadre, l'externalisation ne peut pas jouer un rôle de pression concurrentielle sur les coûts ou de garantie de qualité ; elle ne contribue pas à l'amélioration des modes de contrôle interne.

Le point le plus intéressant concernant la **régie** est l'évolution actuelle de ce type de contrat, qui remet en cause le mode traditionnel de contrôle: les nouvelles "régies forfaitées" associent à la fois un contrôle hiérarchique et une régulation par le marché.

Il apparaît enfin que c'est dans le cas du forfait que le problème du contrôle s'avère le plus crucial. L'analyse qualitative nous a ici permis d'enrichir les résultats de l'analyse quantitative (lien entre l'externalisation au forfait et le caractère structuré du projet) et de comprendre les mécanismes de contrôle dans ce type de contrats. **Plus précisément, la difficulté de mettre en oeuvre un contrôle par le résultat et l'existence de coûts de transaction élevés s'avèrent les principaux freins à la mise en oeuvre de contrats forfaitaires que les entreprises déclarent par ailleurs préférer.**

Nous avons cependant pu constater que certaines entreprises ont réussi à mettre en place des **solutions prometteuses pour contrôler les forfaits** : analyse du besoin structurée et détaillée à l'aide d'outils de prototypage, démarches de plan qualité, métriques de tests. De plus, il apparaît que l'externalisation au forfait peut sous certaines conditions contribuer à améliorer le fonctionnement interne en favorisant un apprentissage sur des modes de pilotage axés sur le contrôle des résultats.

Au total, et prenant du recul sur ces différents éléments, il semble que l'on voit se développer dans les trois situations contractuelles des modes de contrôle mixtes (ou si l'on préfère "complets"), intégrant à la fois des mécanismes de régulation par l'évaluation des comportements et par l'évaluation des résultats. En interne, on voit apparaître des engagements

stricts sur les budgets (régulation par le contrôle des résultats). Le suivi des forfaits intègre quant à lui de nombreuses réunions de point et de suivi en cours de projets ainsi que des plans qualité définissant certaines modalités du comportement du fournisseur. Nous pensons que c'est dans la mise en oeuvre de ces modes de contrôle mixtes, que l'externalisation contribue à développer, que peuvent se situer les voies d'amélioration de la maîtrise de l'activité les plus prometteuses pour l'avenir.

Section 4 - Réflexions sur "Maîtriser l'externalisation"

Au terme de ce long chapitre nous avons pu voir les avantages, inconvénients et dysfonctionnements éventuels de chacune des trois formes contractuelles envisagées, interne, régie et forfait. L'externalisation ne se présente pas obligatoirement comme "la bonne solution" de pilotage ; elle présente des difficultés importantes de mise en oeuvre.

Nous souhaitons dans cette dernière partie, rédigée en forme de conclusion du chapitre, revenir sur la conduite des contrats d'externalisation et prolonger la réflexion sur deux éléments complémentaires :

- dans une logique d'externalisation, peut-on aller plus loin et envisager de se passer de l'informatique interne ?

- que devient le rôle de l'informatique interne dans la relation tripartite qui s'instaure entre l'entreprise (le client), le fournisseur extérieur et l'informatique interne ?

4.1 Peut-on se passer d'un service informatique interne pour le développement d'applications ?

Peut-on se passer totalement d'un service interne pour l'activité de développement d'applications ? Nous avons systématiquement posé cette question à nos interlocuteurs, utilisateurs et informaticiens, qui l'ont souvent trouvée provocatrice (surtout les informaticiens!) et qui y ont répondu - à une exception près - par la négative, en argumentant leur réponse autour de deux grands thèmes :

- l'existence d'un "**noyau minimal**", impérativement conservé en interne pour des raisons essentiellement **stratégiques**,

- la nécessité d'une **protection** de l'entreprise contre ses fournisseurs extérieurs, qui ne peut être assurée que par des spécialistes informatiques internes, et qui répond elle à la fois à des objectifs stratégiques et financiers (limiter les coûts de transaction). **L'informatique interne joue le rôle d'assurance, de garantie** dans un marché où l'évaluation du bien est délicate.

4.1.1 La définition d'un "noyau interne"

Quel est le "noyau minimal interne" ? Reprenons les propos de l'un de nos interlocuteurs, dont nous avons déjà cité certains extraits dans ce chapitre :

"Dans chaque métier, on identifie une partie dont on dit a priori, enfin dont on dit après étude, c'est quelque chose que l'on doit maîtriser en interne. Mais cela ne veut pas dire que l'on doit tout faire en interne. (...) Si l'on fait faire le logiciel, on doit le posséder, on ne doit pas être à la merci de la rupture de continuité d'un sous-traitant quelconque. (...) La plupart du temps, on s'arrange pour que les sous-traitants n'écrivent pas dans des langages connus d'eux-mêmes et de leur concierge. On choisit des standards du marché. Et en général, on met quelqu'un du service sur le coup. De façon à avoir l'option après, même si on ne l'exerce pas, quand il faut faire des modifications, soit de faire en interne, soit de faire faire, par ce fournisseur ou par un autre. Encore faut-il pour cela que l'on connaisse suffisamment ce que l'on a acheté. On garde une maîtrise par un droit de propriété et on s'efforce d'avoir une connaissance suffisante pour pouvoir intervenir, même si tout compte fait, la plupart du temps, pour des raisons de temps, de surcharge, de taille ou autre, on ne le fait pas." (J)

Dans une autre entreprise, nous trouvons un discours assez similaire :

"On doit garder la maîtrise de notre système d'information. J'ai un responsable applicatif qui est un interne. On a sous-traité au forfait mais on a gardé cette personne qui est en tête de pont de l'équipe. C'est elle qui a toute la relation utilisateurs, c'est elle qui juge de l'opportunité, qui discute avec l'utilisateur sur le contenu des modifications, etc., et ensuite passe le dossier à l'équipe de tierce maintenance qui fait un devis. Elle valide le devis et elle recette, elle fait les jeux d'essais, etc. (...) On tenait beaucoup à garder la maîtrise de notre système d'information, c'est un système d'information stratégique pour l'entreprise, donc il n'était pas question de laisser les clés à quelqu'un d'autre sans complètement contrôler. S'il y a un désaccord, si ça ne marche pas, la personne est capable de reconstituer son équipe, ça mettra un peu plus de temps, mais il n'y a pas de problème. Il n'y a pas de perte de compétence." (A4)

Ce qui apparaît dans ces deux témoignages (et dans d'autres que nous ne citons pas mais qui sont convergents), c'est la volonté de préserver ce que nous pourrions appeler "la continuité d'exploitation", c'est-à-dire se donner les moyens de piloter en interne l'activité de développement, même si en pratique celle-ci est largement externalisée. C'est également garder une alternative à l'externalisation, préserver le retour en arrière vers un mode interne, maîtriser directement ce qui est jugé stratégique.

Pour atteindre ces objectifs, un "petit noyau" suffit, comme l'indique notre premier interlocuteur : *"Ce que l'on veut c'est avoir une informatique qui soit à la fois capable de comprendre et maîtriser la petite partie véritablement à laquelle on tient. Quand on a quatre, cinq personnes, cela suffit largement, qui soient toujours conscientes que même s'ils ne font pas eux-mêmes ce n'est pas important. Qu'ils essaient de résister à la tentation de s'agrandir."* (J)

Un autre, partisan de l'externalisation, explique : *"Les responsables de domaine d'applications sont toujours des gens du groupe, les gens qui travaillent sous leurs ordres ne le sont plus".* Il précise : *"Le mode de gestion n'est plus le même. Il faut acheter ce que les autres font mieux que soi. Et se concentrer sur notre métier. Le métier de l'entreprise est de fabriquer et de vendre nos produits, d'élaborer des normes de qualité. Les autres activités ne sont que des fonctions de support par rapport à cette fonction principale."* (F)

Il envisage même - mais il est le seul dans notre échantillon - une informatique sans informaticiens internes, à base de produits standard : *"Dans l'absolu, tout le monde connaît des petites sociétés qui fonctionnent sans informaticiens maison. Pour cela, il faut se limiter aux packages et aux solutions existantes. Dire non seulement je vais prendre le package mais je vais me limiter aux fonctionnalités du meilleur package que j'aurai pu trouver. A partir du moment où on renonce à tout ce qui est spécifique, où on se limite à des solutions standard, rodées, existantes, ce qui compte c'est la vision de l'utilisation de ces packages et la compétence de paramétrage. Un financier peut tout à fait faire du paramétrage de package financier, un industriel du paramétrage de package de production et se faire aider pour*

l'installation, ce sont des sur puissances temporaires. Après tout il n'y a pas nécessité d'avoir un informaticien dans un business, pas nécessité absolue. Il n'y a pas de limite. La limite, c'est le niveau zéro. A condition bien sûr de savoir nouer des relations de longue durée avec le prestataire qui entretient le package, passer de version en version." (F)

Encore faut-il pouvoir se contenter de produits standard - et les identifier sur le marché, les tester, les installer, les interfacer ! Quand bien même ce serait le cas, un autre souci retient les entreprises dans la voie de l'externalisation globale : protéger l'entreprise contre ses fournisseurs extérieurs, c'est-à-dire minimiser les risques et les coûts, jouer le rôle d'assurance et de garantie.

4.1.2 Minimiser les risques et les coûts

Quand les utilisateurs entrent directement en relation avec un fournisseur extérieur et gèrent eux-mêmes cette relation, de façon indépendante de l'informatique interne, cela se passe souvent mal :

"Les utilisateurs avaient choisi un progiciel de façon autonome, mais pour faire les interfaces, ils n'ont pas suivi. Au niveau utilisateur, cela devenait trop technique. Ils voient le progiciel, ils se disent ça nous plaît, ça va, mais ils ne voient pas tout ce qu'il y a derrière.." (C1)

"Il y a des utilisateurs qui ont fait des choses complètement seuls, parce qu'ils avaient envie de le faire, parce qu'on n'avait pas le temps. Ils avaient tellement envie de le faire qu'il valait mieux qu'ils le fassent. Un an après, deux ans après, ils sont venus nous voir, nous ont expliqué "on ne s'en sort pas", et on reprend. C'est difficile..." (B)

"Face aux SSII, les utilisateurs sont démunis. Si on leur met en face quelqu'un d'interne qui suit le projet en back-up, ils n'ont pas une mauvaise image de l'informatique interne. Faire réaliser une application de huit cents jours sur gros système ou sur Oracle/Unix sans avoir aucun repère, pourquoi huit cents jours, pourquoi cinq cent, pourquoi mille, ils se font avoir, on leur a démontré cela." (D)

"Quand le problème est compliqué, les moyens le sont également et ne sont pas, évidemment, maîtrisés par l'utilisateur. On lui parle d'architecture client-serveur, de base de données, de choix de matériel, cela devient très vite quelque chose qui échappe au gestionnaire moyen, même un peu frotté de micro-informatique". (J1)

Le monde de l'informatique est un monde sans pitié ! Vouloir gérer une relation avec un fournisseur informatique sans maîtriser soi-même la technique est un pari dans lequel les entreprises paraissent peu soucieuses de se lancer. **L'informaticien interne, par sa connaissance du métier, va souvent permettre de limiter les coûts de transaction a priori et a posteriori.**

Tout d'abord dans la sélection du partenaire extérieur, dans un contexte caractérisé par une incertitude sur la qualité des fournisseurs et des produits : *"Trouver le bon intervenant, c'est du réseau, de l'expérience positive ou négative..." (B)*, ou encore, sous une forme plus directe : *"Depuis 15 ans que je fais ce métier, je sais quelles sont les sociétés de service qui vous plantent systématiquement, celles qui vous plantent une fois sur deux et celles qui ont tendance à ne jamais vous planter. C'est une réputation sur la base de travaux effectués dans d'autres sites informatiques." (D)*

Ensuite, dans le contrôle de la relation : *"Je crois qu'il faut quand même un spécialiste pour bien contrôler la sous-traitance. Il faut dominer les sujets que l'on sous-traite, c'est indispensable. Les utilisateurs, ce n'est pas leur boulot. Chacun son travail. (...) Toutes les applications un peu stratégiques, qui rentrent dans notre système d'information, passent par nous. Ou en tout cas, on est au moins à un moment ou à un autre dans la boucle." (A3)*

Et enfin, pour gérer les évolutions. Les avis convergent sur ce point : pour envisager l'externalisation sous sa forme radicale, il faut que le système tourne correctement et soit de plus assez stable : *"Pour externaliser, il faut déjà être un peu organisé. Avoir une base un peu saine, une vision, être capable d'exprimer une sorte de cahier des charges global. Il faut être un peu mûr." (H)*

Ainsi, l'informatique interne apparaît comme un relais indispensable entre l'entreprise et les fournisseurs spécialisés en informatique. Nous allons maintenant tenter de détailler quel pourrait être son rôle, dans une perspective d'externalisation.

4.2 Le rôle de l'informatique interne dans la relation tripartite : entreprise - informatique interne - fournisseur extérieur

Les entreprises que nous avons étudiées utilisent quelques règles simples pour organiser la relation entre le client (l'utilisateur), le fournisseur extérieur et l'informatique interne. L'une de ces règles, adoptée par l'ensemble des entreprises, est de placer le fournisseur extérieur pendant l'exécution du contrat sous le contrôle de l'informatique interne, qui assume ainsi une partie de la responsabilité du projet : *"C'est la direction informatique qui est responsable du sous-traitant."* (B)

On se trouve ainsi devant une **double relation de contrôle** dans laquelle l'informatique interne semble jouer le rôle de pivot (figure 9.2).

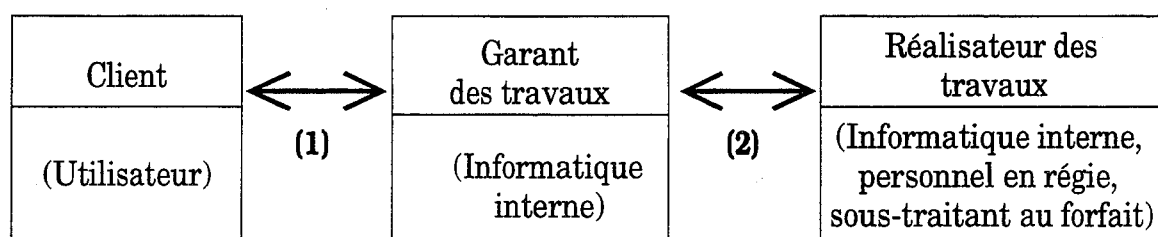


Figure 9.2. La double relation de contrôle

- une première relation (1), lie le client (l'utilisateur) et le "garant des travaux"
- une seconde relation (2), lie à son tour le "garant" et le "réalisateur des travaux"

Nous allons maintenant détailler quel pourrait être, à notre avis, le rôle de l'informatique interne en tant que garant des travaux, en nous appuyant sur les exemples que nous avons pu observer dans les différentes entreprises étudiées et ce dans le sens d'une implication

croissante: information, conseil dans le choix du fournisseur extérieur, garantie technique, responsabilité financière au travers de l'engagement dans son budget... sans oublier cependant de signaler pour finir qu'elle n'est cependant ni seule responsable, ni responsable de tout !

4.2.1 Le "gardien du temple" de l'information

Nous avons vu que, selon les entreprises, les utilisateurs interviennent plus ou moins dans la décision d'externalisation ; dans certains cas elle est initiée par eux, notamment si le projet concerne l'acquisition et l'installation d'un progiciel ; dans d'autres cas cette décision fait l'objet d'une discussion avec la direction générale et les informaticiens ; elle peut enfin être décidée sur un plan technique directement par les informaticiens sur des critères de production. Mais quel que soit le mode de décision, les entreprises ont toutes mis en place des procédures d'information de la direction informatique, destinées à lui permettre d'exercer un contrôle :

"Normalement les utilisateurs passent tous par nous. Il y a un point global, de toute façon, c'est que les achats à un moment donné passent obligatoirement par l'informatique. Les investissements sont obligatoirement signés par l'informatique. Donc à partir de là, il y a un point de contrôle."(E2)

"Le rôle de la direction informatique groupe est d'être un coordinateur des travaux et le contact des SSII". (I1)

L'informatique interne apparaît ici, selon l'expression de l'un de nos interlocuteurs, le "gardien du temple" (F) ; elle garantit une vision globale sur le système d'information et la cohérence des choix, en particuliers techniques ; elle se préoccupe des problèmes d'interfaces des nouvelles applications avec les systèmes existants. Un autre interlocuteur exprime la même idée sous une forme différente : *"L'organisation, les gens ont bougé, il y a eu plusieurs générations en cinq ans mais la direction informatique est restée l'entité permanente, la mémoire, celle qui assure la continuité, le développement linéaire des systèmes d'information."* (H).

4.2.2 Conseil dans le choix du fournisseur extérieur

"Sur un projet quelconque notre rôle c'est d'être aux côtés de l'utilisateur pour l'aider à formaliser son choix, trouver une société... de toute manière le vrai chef de projet, c'est l'utilisateur. Le chef de projet GPAO, c'est quelqu'un de la production, le chef de projet paie c'est quelqu'un de la gestion des ressources humaines, cela ne peut pas être un informaticien. Mais si on fait appel à la sous-traitance, le rôle de la personne de l'informatique interne, c'est d'aider à trouver le meilleur sourcing possible, aider à dire il faut définir le besoin, aider à faire un benchmark..." (F)

Dans ce rôle de conseil se trouve englobée l'analyse précise du besoin et l'utilisation si nécessaire de techniques de maquettage et de prototypage déjà signalées.

4.2.3 Garantie technique

Nous avons longuement détaillé les méthodes d'encadrement des contrats au forfait. Cet encadrement exige une parfaite connaissance technique des processus de développement ; il est assuré par l'informatique interne :

"Nous faisons des cahiers des charges plus structurés, plus formalisés. Il y a une participation forte de l'informatique qui surveille les règles de développement des projets." (I1)

"On a une politique d'externalisation de la programmation. Chaque fois que l'on peut, on fait des cahiers des charges, on fait des appels d'offres et on sous-traite entièrement la partie réalisation. L'informatique interne ne réalise pas, elle n'encadre pas directement les travaux. Elle joue le rôle de maître d'oeuvre vis-à-vis de nos partenaires utilisateurs. Elle recherche et coordonne des entreprises qui font la réalisation." (A1)

4.2.4 Responsabilité financière, engagement sur le budget

Au-delà de la responsabilité technique, les utilisateurs et la direction générale souhaitent souvent que l'informatique interne assure le respect du budget global de l'opération : *"Quand nous sous-traitons au forfait, en principe, il n'y a pas d'avenant. Il y a une surveillance de l'informatique interne qui participe au projet en faisant un suivi de l'avancement" (E1)*, voire

même s'engage sur ce budget : *"Nous avons fait un forfait utilisateurs, dans lequel nous avons englobé les prestations de notre fournisseur et les nôtres. Si le fournisseur dépasse parce qu'on lui a demandé des modifications fonctionnelles, cela va se répercuter sur les utilisateurs. Si c'est de notre fait, sur des tâches dont nous sommes responsables, et qu'il y a eu des répercussions chez le fournisseur, ce sera pour nous. Nous avons des comptes d'exploitation et ça ira dans notre ligne de résultat."* (K)

4.2.5 Mais attention, ni unique responsable, ni responsable de tout

Gardien du temple, conseil, responsable technique, responsable financier... au total le risque finit par être que l'informatique interne forme un "écran" entre le client (l'utilisateur) et le fournisseur extérieur. Écoutons par exemple le discours d'un responsable interne des études d'une entreprise qui pratique - entre autres - l'externalisation au forfait et dont nous pourrions dire qu'il est "victime de son succès" : *"Le problème, aujourd'hui, c'est que l'on prend tout en main. Tout ! Y compris presque le chef de projet utilisateur. On est souvent à l'initiation du projet, on est à la définition du besoin, on est à la rédaction du cahier des charges, on est à la maîtrise d'oeuvre. Et on est à l'installation. On est partout. Ça peut créer quelques difficultés, parce que si les délais sont un peu dépassés, si les coûts ne sont pas tellement respectés, l'utilisateur a beau jeu de..."* (A1)

Et encore : *"C'est un problème dans l'entreprise, le positionnement, le rôle des utilisateurs. Il devrait y avoir des maîtrises d'ouvrage bien claires, bien fortes, qui prennent bien en charge leurs responsabilités, qui définissent des chefs de projet utilisateurs. Il y a un transfert total de responsabilité qui n'est pas bon."* (A1)

Notre objectif n'est pas ici d'aborder en détail les responsabilités de l'utilisateur, qui ont été par ailleurs récemment décrites avec précision dans le très bon ouvrage consacré à la "maîtrise d'ouvrage" par l'Association Française d'Audit Informatique (Afaï, 1993) ; qu'il nous suffise de rappeler qu'elles doivent être assumées, au même titre que celles de l'informatique interne et du fournisseur extérieur, pour que la "double relation de contrôle" que nous avons définie puisse fonctionner de façon satisfaisante.

Les entreprises ont encore vraiment besoin de leur informatique interne ! Comme nous l'explique un de nos interlocuteurs : *"Oui, j'ai pensé à l'externalisation, mais alors, je donne quelque chose en état, que cela tourne. Le jour où l'on aura une configuration correcte, où l'on aura mené nos principaux chantiers, où l'on aura maîtrisé l'informatique, quand ce sera en place, pour continuer à gérer tout cela, et faire évoluer, pourquoi pas. Mais même comme cela je ne conçois pas de ne pas garder un noyau interne pour ne pas se faire embarquer... pour pouvoir décider éventuellement."* (J)

On voit bien apparaître ici ce qui nous semble être les deux principales limites de l'externalisation sur le plan du contrôle : tout d'abord la difficulté de gérer avec un partenaire extérieur une situation évolutive et confuse ("je donne quelque chose en état, que cela tourne"); ensuite la nécessité de garder la maîtrise interne et de limiter le risque de dépendance ("pour ne pas se faire embarquer, pour pouvoir décider").

Annexe : Guide d'entretien semi-directif

I - Caractéristiques de l'entreprise

Type d'activité. Part relative dans le CA. Résultats présents et passés
Effectifs et évolution
Facteurs clé de compétitivité pour le secteur et l'entreprise
Organisation (centralisée / décentralisée)
Organisation informatique (centralisée / décentralisée)

II - Rôle de l'informatique

Quel est selon vous le rôle de l'informatique dans l'entreprise ?
Quelles sont les missions de l'informatique ? Qu'en attend la direction générale ?
Comment considère-t-on l'activité de développement d'applications spécifiques ?

Applications de gestion dans l'entreprise

Etat général des logiciels : ancienneté, niveau de satisfaction
Liste des principales applications informatiques existantes
Parc applicatif
Applications de gestion importantes (rapidement)
Applications nouvelles mises en service depuis 2 ans
Applications en attente. Pourquoi ?

III. Choix d'externalisation

Décision

Quelle est la politique aujourd'hui en matière d'externalisation (de sous-traitance) ?
Comment se fait le choix de l'appel à un fournisseur extérieur ?

- Qui décide (DG, utilisateurs, informaticiens) ?
- Y a-t-il examen de plusieurs solutions et mise en concurrence systématique de l'informatique interne (progiciel, source interne, source externe spécifique) ?
- Y a-t-il appel d'offres ouvert ?
- Qui tranche ? Quels sont les critères utilisés ?

Les utilisateurs doivent-ils consulter obligatoirement l'informatique interne ?

Modalités et contrôle

Quelle forme de sous-traitance utilisez-vous ? (régie/forfait) Dans quelles circonstances ?

Régie

Dans quelles circonstances utilisez-vous la régie ?

- qu'attendez-vous de la régie ?
- quels types de profils recherchez-vous ?
- pour quels types de projets utilisez-vous la régie ?

Que faites-vous faire aux équipes en régie ?

- utilisez-vous la régie de préférence à certaines étapes du développement ?
- faites-vous une différence dans l'affectation des tâches entre le personnel interne et le personnel en régie ?

Etes-vous satisfaits de la régie ? Pourquoi ?

Comment contrôlez-vous le travail en régie ?

- utilisez-vous la "régie forfaitée" ?

Forfait

Dans quelles circonstances utilisez-vous le forfait ?

- qu'attendez-vous du forfait ?
- pour quels types de projets utilisez-vous le forfait ?
- utilisez-vous le forfait de préférence à certaines étapes du développement ?

Etes-vous satisfaits du forfait ? Pourquoi ?

Comment contrôlez-vous les travaux réalisés au forfait ?

- technique de définition du cahier des charges ; utilisation du prototypage
- plan qualité

Comment se traitent les ajustements en cours de projet ? (évolution des spécifications, dépassements par rapport au budget)

Fournisseurs extérieurs

Quels sont vos principaux fournisseurs extérieurs ?

Pourquoi avez-vous choisi de travailler avec eux ?

Pensez-vous que vous pourriez vous passer de l'informatique interne ? Quel est le "noyau minimal interne" ? (question souvent posée en fin d'entretien)

IV - Description du service informatique interne

Taille (effectifs internes / externes en régie)

- globale, - par sous-ensemble (études, exploitation, etc.)

Situation de l'informatique dans l'organigramme général de l'entreprise

Organisation des études (organigramme)

Principales tâches du service études

Capacité de production :

- Etes-vous en sur/sous capacité ? Volume de projets en attente ?

Quelle est votre position par rapport aux technologies (avant-garde, suiveur)

- existence d'une cellule de test des nouvelles technologies
- utilisation AGL? Fonctions mises en oeuvre ?
- utilisation d'outils de prototypage ?

Personnel informatique

- "profil" des informaticiens internes ?
- les informaticiens sont-ils des spécialistes du métier de l'entreprise ?
- existe-t-il des passerelles entre l'informatique et les autres services en termes de carrière ?

Pouvez-vous décrire les relations entre les informaticiens et les utilisateurs ?

V. Contrôle de l'informatique

L'informatique interne a-t-elle un budget ? Sur quelles bases.

En règle générale, les budgets sont-ils respectés ? Et les délais ?

Existe-t-il un système de facturation des développements sur les autres services ?

- sur quelles bases ?
- trouvez-vous le système juste ?

VI. Système de planification et de choix des applications

Décision de lancement

Comment est piloté le choix des nouvelles applications informatiques ?

Qui décide de lancer une nouvelle application ? Sur quel budget ?

Qui est responsable d'une nouvelle application (utilisateur/informaticien)

Existe-t-il une procédure de hiérarchisation des demandes de nouvelles applications ?

- Quels sont les critères formels utilisés ? Les critères informels ?
- Existe-t-il une procédure d'évaluation des nouveaux projets (estimations de rentabilité *a priori*, contrôle après mise en oeuvre)

Conclusion générale

Parvenu à l'issue de ce travail, nous allons tenter d'en dégager les principaux apports, en rappeler les limites et identifier les perspectives qu'il permet d'ouvrir.

Notre objectif quand nous avons entrepris cette recherche était **d'analyser et de comprendre les choix d'externalisation des entreprises, concernant l'activité de développement d'applications**. Notre intérêt pour cette question avait deux origines. Tout d'abord la question de l'externalisation de l'informatique venait d'être projetée sur le devant de la scène au travers de quelques contrats fortement médiatisés et qui semblaient faire école. La presse professionnelle faisait écho de nouvelles solutions de gestion de l'informatique, à base d'externalisation, dans lesquelles le poids des acteurs spécialisés du marché - les SSII - serait dominant, et qui pouvaient concerner l'ensemble des activités informatiques. D'un autre côté, les quelques études disponibles traitaient principalement des activités d'exploitation. Nous avons donc choisi de nous intéresser à la question de l'externalisation de l'activité de développement d'applications, sur laquelle très peu de travaux de recherche avaient été effectués et qui paraissait moins facile à résoudre. En effet, les modalités que pouvait prendre l'externalisation de cette partie de l'activité informatique n'étaient pas claires, pas plus que les risques qu'elle comportait pour les entreprises : au fond, nous nous demandions si la maturité industrielle du "produit" (le logiciel spécifique) était suffisante pour envisager l'externalisation avec profit, si les entreprises étaient à même de piloter cette délégation tout en gardant la maîtrise de leurs systèmes, si elles ne prenaient pas des risques stratégiques en confiant cette tâche à l'extérieur et si la qualité des prestations offertes par les fournisseurs pouvait être garantie.

Nous avons donc **cherché dans ce travail à répondre à trois questions** :

1. L'activité de développement d'application est-elle trop stratégique pour être externalisée ? (approche stratégique de la ressource).
2. Y a-t-il, concernant le développement d'applications, un véritable avantage lié à la spécialisation industrielle, qui justifie de confier cette activité aux professionnels du secteur informatique ? (approche industrielle).

3. L'entreprise est-elle à même de gérer la relation avec les fournisseurs extérieurs - ou, plus précisément, l'externalisation ne risque-t-elle pas d'entraîner des coûts supplémentaires prohibitifs de gestion de la relation ? (approche de contrôle).

Pour tenter de répondre à ces questions, nous avons choisi d'adopter successivement trois points de vue (stratégique, industriel, de contrôle) et **d'analyser l'activité d'applications selon chacune de ces approches théoriques**. Nous avons ainsi été amenés à faire le point des recherches et nous sommes efforcés de définir "l'état de l'art" sur la question de l'intérêt stratégique du développement d'applications, ainsi que sur les aspects liés au processus de développement, en étudiant en particulier les évolutions technologiques récentes. Nous avons également cherché à approfondir la réflexion sur le sujet encore mal exploré du contrôle de l'activité, en remontant aux sources des difficultés de contrôle par l'analyse des caractéristiques du "produit" (le logiciel spécifique) et des transactions (le développement par des informaticiens, -internes ou externes- pour le compte des utilisateurs). La confrontation des approches théoriques avec notre objet d'étude nous a ainsi permis d'identifier un ensemble de **facteurs de contingence**, que nous avons testés lors de la recherche empirique.

Celle-ci comprend deux volets. Tout d'abord nous avons senti la nécessité d'une **phase descriptive des comportements**, visant à faire le point sur les pratiques actuelles. Pour cela, nous avons collecté l'information sur les choix d'externalisation effectués par douze entreprises dans le cadre de 60 projets de développement d'applications spécifiques menés dans les deux dernières années. Nous avons à chaque fois demandé de préciser les choix effectués pour les étapes d'analyse du besoin, de conception et de réalisation. Le second volet de l'étude empirique a été **l'identification des déterminants des comportements** ; l'objectif était de tester dans quelle mesure les facteurs identifiés au travers de l'approche théorique pouvaient s'appliquer à notre domaine précis de recherche. L'étude quantitative met en évidence sur la population étudiée l'existence de liens entre les caractéristiques des projets de développement et les choix d'externalisation ; l'étude qualitative, résultant d'entretiens menés auprès de gestionnaires et d'informaticiens de ces entreprises, permet de compléter et de nuancer le portrait des mécanismes de choix d'externalisation.

Avant de rappeler les résultats de l'étude empirique, il nous semble important d'en souligner les principales limites. Tout d'abord, il s'agit d'une étude exploratoire, menée auprès d'un petit nombre d'entreprises ; cette limite se retrouve également dans l'étude quantitative sur les projets, où l'objectif a été d'apprendre en favorisant la diversité des situations et non pas de viser à une représentativité de l'échantillon. De plus, nous avons restreint nos recherches empiriques au cas des entreprises de moyenne et grande taille, évoluant dans des secteurs où l'informatique, même si elle est une fonction de support indispensable, ne peut pas être qualifiée de "totalement stratégique". Le problème de l'externalisation se posait à notre avis en priorité pour elles : l'investissement dans l'informatique ne présentant pas un caractère stratégique, on pouvait envisager d'appliquer à l'égard de l'activité de développement d'applications une logique économique, favorable en règle générale à une spécialisation des acteurs sur le marché.

Quels résultats avons nous obtenus ?

L'étude quantitative a permis de clarifier l'état actuel des comportements. Nous en soulignons ici trois éléments saillants :

- Tout d'abord, nous avons constaté une différenciation très marquée des choix selon les étapes du processus de développement. De façon globale, l'étape d'analyse du besoin est majoritairement menée en interne, l'étape de conception fait l'objet de solutions mixtes interne/régie ou interne/forfait, l'étape de programmation est très largement externalisée, soit en régie, soit au forfait.

Nous ne saurions trop insister sur cette différenciation par étape, qui va dans le sens d'une **différenciation des choix selon la nature des tâches au sein du processus de développement**. Cette différenciation va à l'encontre du discours habituel sur l'externalisation du développement d'applications, où l'on se situe au niveau global du projet ("ce projet a été fait en régie", "ce projet a été sous-traité au forfait") sans prendre la peine de préciser quelle étape du projet est visée par l'externalisation.

- Ensuite, nous avons pu élaborer à partir des données collectées une typologie des choix en six groupes, correspondant à **six schémas de comportement d'externalisation** distincts sur les trois étapes du processus étudiées : le "forfait global", les "forfaits de conception et de réalisation", les "régies d'appoint", les "régies de conception et de réalisation", la "dominance interne" et le "tout interne". De façon globale, nous avons pu constater que les entreprises n'ont pas adopté une solution privilégiée et utilisent chacune, pour la même activité de développement d'applications, différents schémas de comportement.

- Enfin, **l'externalisation en régie et au forfait semblent être influencées par des facteurs différents** : concernant la régie, l'étude quantitative met en évidence l'importance de facteurs opérationnels (ajustements charge/capacité, délais) ; concernant le forfait, on constate un lien entre la nouveauté technologique et le degré de structure des projets et une externalisation plus marquée.

L'étude des facteurs déterminant les choix d'externalisation a été prolongée dans l'analyse qualitative. A l'issue de nos travaux, nos réponses aux trois questions posées sont les suivantes.

- **Aspects stratégiques**

L'approche stratégique de l'externalisation (théorie de la ressource) recommande d'identifier les "ressources et compétences de base" de l'organisation, de développer ces ressources en interne et de privilégier l'externalisation des ressources secondaires pour permettre une meilleure focalisation vers les objectifs stratégiques. Dans cette approche, l'externalisation des ressources secondaires a cependant deux limites : l'entreprise doit garder la maîtrise de la ressource, en pilotant en interne les phases critiques ; elle doit également maintenir en interne une "compétence architecturale", c'est-à-dire la capacité à faire le lien entre ses propres besoins et l'offre technologique.

Nous nous étions placés d'emblée, lors de la sélection des entreprises, dans des secteurs où l'activité de développement d'applications est globalement perçue comme une activité de support (ressource secondaire), contexte *a priori* favorable à l'externalisation. Cependant nous faisons l'hypothèse que des **projets**, importants pour l'organisation, ou des **étapes du**

processus, jugées critiques, pouvaient être considérés par l'entreprise comme faisant partie des ressources ou compétence de base et à ce titre internalisées. Dans ce contexte, rappelons-le limité par le choix de secteurs où l'informatique n'est pas *a priori* stratégique, nos résultats sont les suivants :

- L'analyse des entretiens révèle que **nos interlocuteurs n'émettent pas de réserve sur un plan stratégique à l'externalisation de l'activité de développement, à l'exception cependant de l'étape d'analyse du besoin**, qui est celle où se jouent à la fois la maîtrise de la ressource et le maintien en interne de la compétence architecturale. De façon plus générale, nous avons constaté que nos interlocuteurs font bien le lien entre l'importance stratégique d'une activité (ici même d'une *étape* et de *tâches* au sein d'un processus de production) et leur comportement d'externalisation, confirmant ainsi l'application de la théorie de la ressource à ce cas particulier.

- **Nous n'avons constaté aucune différence dans les pratiques d'externalisation en fonction de l'importance stratégique accordée au projet.** Ce résultat peut paraître surprenant ; nous reconnaissons qu'il doit être considéré avec prudence et qu'il serait nécessaire de le confirmer par une étude ciblée sur la définition et l'analyse du caractère stratégique des projets au travers de différents critères, là où nous avons demandé un jugement global à nos interlocuteurs (estimation du caractère stratégique sur une échelle de Lickert). Il reste que cette constatation qui traduit à notre avis le fait que, même sur des projets stratégiques, les entreprises envisagent de déléguer la conception et la réalisation de l'application à l'extérieur - pour peu que cette délégation puisse s'effectuer dans des conditions de contrôle satisfaisantes.

- **Aspects industriels**

Selon l'approche industrielle, un fournisseur professionnel a de grandes chances de bénéficier d'avantages liés à la spécialisation de ses activités, spécialisation qui lui permet de proposer des prix inférieurs à ceux de l'organisation interne. Les principaux déterminants de cette avantage sont les volumes et flux de production (économies d'échelle) et la technologie (effet d'apprentissage). Nous avons pu montrer dans l'analyse théorique que les technologies de développement d'applications ont atteint actuellement un stade de rupture, où les anciens

modèles et outils de développement sont remis en cause, au profit d'une organisation plus industrielle du processus, appuyée sur de nouveaux outils. La situation française actuelle nous place au début de la vague de diffusion de ces nouvelles technologies, déjà bien adoptées par exemple par les équipes américaines. Notre hypothèse à l'issue de l'analyse théorique était que les choix d'externalisation étaient liés à trois groupes de facteurs industriels : ceux liés aux volumes et flux de production (à court terme : délais et ajustement charge / capacité, à long terme : anticipations sur la pérennité de la demande), le facteur technologique et l'intégration des nouveaux projets au système d'information. Nous avons testé les différents facteurs issus de l'approche industrielle. Nos résultats montrent que :

- Les facteurs **opérationnels** liés aux volumes et flux de production (**ajustement charge/capacité, délais**) **influent directement sur les choix d'externalisation**. Plus précisément, ils sont au cœur de l'arbitrage interne / régie. Nous avons retrouvé là, avouons le, des éléments bien connus par ceux qui s'intéressent à ce domaine d'activité. On peut cependant noter la tendance actuelle des entreprises à gérer les ajustements charge/capacité à court terme et de façon très pointue, ce qui change le mode d'utilisation de la régie, traditionnellement plus axée vers des contrats à long terme.

- Plus intéressant est à notre avis le **mouvement de fond que nous avons observé en faveur de politiques "d'évitement de capacité"**. Nous pensons que l'incertitude sur la pérennité de l'activité de développement d'application explique en grande partie le refus d'investissement en personnel interne manifesté par dix entreprises sur les douze étudiées. Ces entreprises n'ont cependant pas enclenché de politiques de désinvestissement, à l'exception de l'une d'entre elles, qui a choisi de sous-traiter certaines maintenances dans l'objectif de reclasser une partie du personnel informaticien à l'extérieur.

- **L'hétérogénéité et la complexité croissante des technologies de développement jouent également en faveur de l'externalisation**, mais semblent être liés au choix de **contrats au forfait**. Les entreprises que nous avons étudiées, tout en souhaitant bénéficier des nouvelles technologies, se montrent réservées quant à leur investissement dans la connaissance interne des technologies. Celui-ci est très ciblé et ne concerne que les technologies directement utilisables à court terme et de façon extensive par l'entreprise. Pour des projets mettant en

oeuvre de nouvelles technologies, les entreprises reconnaissent en règle générale l'avantage de faire appel à un fournisseur extérieur spécialisé.

- Sur un plan industriel, la principale limite à l'externalisation semble être la **difficulté d'intégration** des nouvelles applications aux systèmes d'informations existants. Cet obstacle ne concerne pas les projets portant sur des applications relativement "isolées" dans le système d'information. Il est contourné lorsque l'entreprise procède au renouvellement complet d'un "bloc" d'applications (le système d'information de production, le système d'information commercial ...). Enfin, dans plusieurs cas, nous avons observé une répartition des tâches entre les fournisseurs extérieurs et l'informatique interne : en externe la partie "nouveaux développements" du projet, en interne, les interfaces.

- **Aspects liés au contrôle de l'activité**

Nous avons mené l'analyse des aspects liés au contrôle de l'activité en cherchant à appliquer deux approches théoriques, celle de la théorie de l'agence et celle des coûts de transaction. L'étude des attributs des produits et celle des caractéristiques des transactions expliquent les racines du problème de contrôle qui se pose concernant l'activité de développement d'applications : le logiciel spécifique est un bien difficile à définir *a priori*, les adaptations en cours de développement sont souvent nécessaires, le résultat dépend du contexte d'utilisation ; les transactions sont quant à elles souvent marquées par une asymétrie d'information entre le fournisseur et son client, des délais assez longs et un risque potentiel élevé.

Notre hypothèse était que les choix d'externalisation étaient liés aux possibilités et aux contraintes de contrôle liées à chaque mode d'organisation (interne, régie, forfait). Nous avons donc cherché à évaluer les performances des différents modes d'organisation sur le plan du contrôle et à identifier les causes d'échec éventuel des uns ou des autres.

- Un des principaux apports de cette thèse est sans doute de montrer le **lien entre les comportements d'externalisation et le fonctionnement du contrôle interne**. Nous avons identifié trois situations différentes de contrôle interne, dans lesquelles le recours à l'externalisation joue un rôle différent. Le fait de savoir qui décide l'externalisation et avec

quels objectifs (coût, "qualité globale" ou critères opérationnels) s'avère en particulier déterminant.

- Concernant la **régie**, nous avons pu observer la tendance à la mise en place par les organisations, en plus du contrôle hiérarchique traditionnel, de modes contrôle appuyés sur un engagement du fournisseur sur un résultat : plusieurs entreprises pratiquent ainsi une "régie forfaitée", dans laquelle le fournisseur travaille sous les ordres et sous la responsabilité de l'informatique interne mais s'engage néanmoins pour chaque prestation sur un délai et un volume global d'intervention. La tolérance dans le respect des obligations reste cependant plus forte que dans le cas des contrats au forfait, même si elle tend à se réduire.

- Enfin, un aspect important de notre analyse du lien entre les possibilités de contrôle et les pratiques d'externalisation concerne les contrats au forfait. **Le problème du contrôle semble être l'obstacle principal à une externalisation au forfait, que la plupart de nos interlocuteurs appellent par ailleurs de leurs vœux.**

Nos interlocuteurs soulignent en particulier la difficulté qu'ils éprouvent à définir et à évaluer le résultat de la prestation et à éviter les comportements opportunistes de la part des fournisseurs spécialisés. Par ailleurs, nous avons souvent été frappés par les difficultés qu'éprouvent les gestionnaires à préciser leurs besoins, à les conceptualiser et à faire le lien entre ces besoins et les outils informatiques. Nous avons cependant pu montrer au travers de certains exemples qu'il existe des moyens pour réduire ces difficultés, passant par la mise en oeuvre de politiques internes de sous-traitance au forfait. Les méthodes et outils de pilotage des projets développés dans le cadre de ces politiques d'externalisation sont centrés sur la qualité et la définition des résultats. Ils améliorent la maîtrise des forfaits ; mais ils sont également source d'apprentissage dans la gestion des projets réalisés en interne ou en régie. L'entreprise peut en effet bénéficier de l'expérience des professionnels de l'informatique, dont l'intérêt est de développer ces méthodes et outils de pilotage pour attester de la qualité de leurs prestations, et enrichir ainsi les modes de contrôle hiérarchiques usuels.

Peut-on, à l'issue de ce travail, dépasser le cadre descriptif et analytique que nous nous étions fixé et faire des propositions concernant les décisions d'externalisation ?

Ces choix apparaissent à l'évidence extrêmement contingents ; cependant nous pensons que certaines recommandations générales peuvent s'appliquer.

Tout d'abord, nous conseillerions aux entreprises de faire de la réflexion sur l'externalisation des développements spécifiques **un des points centraux de leur politique de gestion de l'activité de développement**, un peu comme elle l'ont fait ces dernières années pour l'acquisition de progiciels. Il s'agit ici d'élargir le champ des possibles et d'être à même, à tout moment, de faire en interne ou en régie, d'acheter, de piloter des forfaits dans des conditions et avec des chances de succès acceptables, voire de combiner ces solutions au sein d'un même projet. Nous pensons que les entreprises ont tout à gagner de cette ouverture des choix, tant sur le plan de la réduction des coûts de développement et de l'accès aux nouvelles technologies que sur celui de la réactivité.

Quand elles externalisent, nous pensons qu'elles devraient chercher à **privilégier vraiment le forfait**. C'est-à-dire qu'elles ont pour la plupart d'entre elles à franchir la distance qui sépare les déclarations d'intentions ("Le forfait, c'est la solution que je préfère") des décisions effectivement prises ("En pratique, je fais beaucoup plus appel à la régie").

Il leur faudrait donc savoir **mieux reconnaître les projets et les étapes du processus** sur lesquels elles peuvent envisager de commencer à pratiquer le forfait. En pratique, nous avons pu voir que les projets très structurés et à forte composante technologique sont de bons cadres pour la mise en oeuvre de forfaits sur les étapes de conception et de réalisation. Afin de limiter les risques, le choix de petits projets non stratégiques s'impose, au moins en phase d'apprentissage de la mise en oeuvre de forfaits.

De même, il leur faudrait savoir **garder en interne** au minimum le "noyau" permettant de préserver la compétence architecturale, c'est-à-dire la capacité à identifier les opportunités technologiques et à traduire de façon précise les besoins de l'entreprise en spécifications techniques. Nous pensons que l'adoption de techniques de prototypage et l'utilisation des *upper case* (outils d'aide à la définition du besoin des ateliers de génie logiciel) est un moyen d'aller

vers cet objectif. De même, dans l'état actuel du marché et des risques qu'il présente (opportunisme de certaines SSII, insuffisance des mécanismes de régulation de la profession du type certificat d'assurance qualité ou même "conseil de l'ordre" pour suivre l'exemple de la profession médicale), il convient d'être prudent dans la voie de l'externalisation. Nous avons été sur ce point étonnés (voire même parfois inquiétés) par le fait que les entreprises étudiées semblent ne pas différencier leurs choix en fonction du critère de l'importance stratégique du projet. Il nous semble que les entreprises devraient conserver un engagement important de leur personnel interne dans les projets stratégiques, comme elles le font pour les projets qui mettent en jeu des technologies importantes et réutilisables par l'entreprise.

Enfin, **la maîtrise des forfaits nous paraît être une réelle compétence de base pour les organisations**. Cette maîtrise ne s'improvise pas. Elle suppose au contraire un travail et un apprentissage coordonnés, que peu d'entreprises véritablement entrepris et qui correspond à la redéfinition de l'ensemble du processus de développement. Cet apprentissage a plusieurs dimensions : la connaissance du marché, c'est-à-dire la multiplication des réseaux d'information sur les fournisseurs ; le développement des compétences juridiques (maîtrise des clauses contractuelles, connaissance de la jurisprudence, des procédés d'expertise en cas de litige *avant* de se trouver confronté soi même à la situation - et surtout pour éviter de s'y trouver confronté) ; la capacité à évaluer les outils de développement utilisés par le fournisseur (quel atelier de génie logiciel utilise-t-il, quels sont ses méthodes de conception et de réalisation) ; le développement de plans qualité, de procédures et de métriques de tests. La plupart de ces compétences sont disponibles sous forme d'expertise ; encore faut-il développer le réseau d'experts autour de l'entreprise et intégrer une partie de la connaissance en interne.

Nos propositions s'adressent également à la **profession informatique**. Pour pouvoir jouer un rôle de plus en plus important dans le développement d'applications, il nous semble que devraient être développés au niveau de la profession des mécanismes d'assurance qualité (certification, label, propositions de plan qualité) et des règles de déontologie contraignantes destinées à compenser l'incertitude sur la qualité qui est une composante intrinsèque de ce type de production. Des travaux comme ceux de l'AFNOR ("Qualité et ingénierie du logiciel",

1993), ceux de l'IEEE (norme 1061 : Software Quality Metrics Metrology, 1992), le recueil ISO 9000 sur les normes internationales pour la gestion de la qualité et la réflexion de la commission des communautés européennes sur la responsabilité du prestataire de services sont des avancées importantes en ce domaine.

Enfin, nous souhaitons souligner ici deux pistes de recherche complémentaires soulevées par notre travail et qu'il nous semble important de prolonger par des travaux futurs.

La première piste de recherche concerne le problème du contrôle de ce type d'activité. Au total, et en prenant du recul sur cette problématique de contrôle, l'activité de développement d'applications apparaît au travers de notre étude comme un exemple frappant des **difficultés de pilotage de transactions portant sur des biens dont la qualité est difficilement attestable a priori**, rejoignant les "voitures d'occasion" de Akerlof (1970), les oranges de Barzel (1982) et les professions de service comme les médecins ou les avocats. Nous espérons avoir ainsi contribué à l'analyse des mécanismes intervenant sur ce type de transactions et de marchés, qui se situent aux limites des modes de régulation économiques classiques. Nos apports concernent à notre avis principalement deux aspects :

- tout d'abord, ces mécanismes ont été étudiés dans le cadre relativement peu exploré des produits à forte composante technologique. Nous pensons que la technologie, surtout dans les phases de renouvellement ou de bouleversement des procédés, peut être un facteur d'opacité sur la qualité des biens. Ces mécanismes ont été analysés par Lambert (1993), dans le cas du transfert de technologie : "Certains auteurs, spécialisés dans le problème des transferts de technologie, pensent que l'information sur les réelles potentialités de la technologie offerte constitue la barrière principale aux mécanismes de transfert (...) Il semble qu'une forme plus sérieuse de barrière provienne de la traduction d'un savoir-faire technologique dans un langage compréhensible par l'utilisateur." (p. 62, souligné par nous). L'auteur souligne que les efforts faits pour rendre la technologie plus explicite sont "la source de coûts de transaction la plus importante". Le même handicap peut affecter à notre avis l'externalisation de produits dont la

technologie est abstraite ; nous pensons qu'il y a là une piste de recherche intéressante à approfondir.

- ensuite, il nous semble que les difficultés de pilotage liées au problème de définition du résultat et d'évaluation de la qualité *a priori* ne peuvent pas être résolues par des solutions de contrôle unidimensionnelles ("le marché" ou "la hiérarchie" ou encore "le contrôle du comportement" ou "le contrôle des résultats"). L'un des résultats principaux de cette recherche nous semble être de montrer la généralisation de **formes mixtes** de pilotage, intégrant différents mécanismes de régulation et **s'appliquant aux différentes formes contractuelles**. Nous avons ainsi pu constater le développement, en interne, de mécanismes de contrôle des résultats et celui, dans les contrats au forfait, de procédures de suivi et de pilotage des comportements . Il nous semble que l'étude de ces modes de pilotage mixtes est une voie de recherche intéressante à poursuivre, dans la mesure où elle peut permettre de gérer les nouvelles formes de relations complexes entre entreprises, largement appuyées sur l'externalisation de certaines activités, du type de l'organisation en réseaux.

Enfin, ce travail nous engage à une **réflexion sur le devenir des directions informatiques internes**. Il nous semble que plutôt que de brandir l'externalisation comme une alternative radicale et une menace ("Si le logiciel ne baisse pas en coût, et ne monte pas en qualité, vous serez, vous directeur informatique, un de ces jours, outsourcé"), il faut l'envisager comme un choix complémentaire possible et une opportunité. L'externalisation de certains aspects du développement d'applications permet de mettre en évidence des économies potentielles ; elle oblige l'informatique interne à clarifier sa prestation et introduit une base de comparaison permettant d'évaluer la performance interne.

Dans ce contexte, nous pensons que la **comparaison au marché peut être un moyen d'améliorer la relation entre l'informatique interne et les utilisateurs**, en supprimant certaines causes de friction : dès lors qu'elle est mise en concurrence, l'informatique interne ne peut plus être perçue comme un service privilégié s'appuyant sur son expertise technique pour éviter l'évaluation ; les soupçons sur sa loyauté peuvent être écartés du fait de la contrainte qui s'impose à elle de satisfaire la demande ou d'être marginalisée.

L'analyse de Mintzberg sur les détenteurs de **compétences spécialisées** dans l'organisation apporte ici un éclairage extrêmement intéressant. Mintzberg (1986) identifie deux groupes utilisant ce système d'influence dans l'organisation, les "opérateurs spécialisés" (les experts) et le "personnel de soutien logistique qualifié". Contrairement aux opérateurs spécialisés, le personnel de soutien logistique "offre à l'organisation des services plutôt fragiles" (puisque'ils peuvent être achetés facilement en dehors de l'organisation), n'est pas impliqué dans l'analyse en tant que telle mais est plutôt concerné par les applications de l'analyse" et "traite des changements organisationnels". De ce fait, "pour le personnel de soutien logistique qualifié, la collaboration est très importante" (p. 205, souligné par nous). L'influence du groupe des personnels de soutien logistique ne peut par conséquent s'exercer que dans une perspective de collaboration étroite et de convergence d'objectifs avec l'organisation, ce qui les place dans une situation exactement opposée à celle des opérateurs spécialisés.

Nous pouvons ainsi réinterpréter, à la lumière de notre problématique d'externalisation, la proposition de Lesca (1990), qui plaide pour la définition d'une "direction des ressources d'information" occupée par "un homme d'interaction et de communication". Cette perspective, partagée par un ensemble toujours plus nombreux de chercheurs en systèmes d'information (Applegate et Elam, 1992, Feeny et al., 1992, Cash et al., 1992), éloigne la vision de l'informaticien interne "opérateur spécialisé" pour la remplacer par celle de "personnel de soutien logistique qualifié" - ou, pour utiliser une appellation plus positive, "**d'expert de l'application des technologies informatiques à l'entreprise**". Cet expert serait alors préoccupé par la recherche des meilleures technologies et produits disponibles, l'élaboration de réseaux avec les professionnels du secteur informatique et l'application de ses compétences à l'organisation, en coopération avec les membres des autres fonctions.

Et si une pratique ciblée et maîtrisée de l'externalisation, loin d'être une punition affligée à un service informatique interne jugé inefficace, était une opportunité pour les responsables de systèmes d'information de gérer cette évolution, tout en bénéficiant des progrès de l'industrialisation du logiciel et en améliorant les modes de contrôle interne ?

BIBLIOGRAPHIE

- Ackoff, R.L.**, *Management Misinformation Systems*, Management Science, décembre 1967, p. 147-156.
- AFAI** (Association Française de l'Audit et du conseil Informatique), *Maîtrise d'ouvrage de projet de système d'information*, AFAI, Paris, 1993.
- Agresti, W.W.**, *New Paradigms for Software Development*, IEEE Computer Society, Library of Congress, New York, 1986.
- Akerlof, G.A.**, *The Market for "Lemons" : Quality Uncertainty and the Market Mechanism*, Quarterly Journal of Economics, vol. 84, 1970, p. 488-500.
- Alavi, M.**, *An assessment of the Prototyping Approach to Information Systems Development*, Communications of the ACM, vol. 27, n° 6, juin 1984, p. 91-98.
- Alchian, A.A., Demsetz, H.**, *Production, Information Costs, and Economic Organization*, The American Economic Review, vol. 62, 1972, p. 777-795.
- Allen, B.**, *Make information services pay its way*, Harvard Business Review, janvier-février 1987, p. 57-63.
- Allen, W.H.**, *Successfully Managing the Multivendor Environment*, Journal of Information Systems Management, hiver 1985, p. 13-20.
- Anderson, E.E.**, *Choice Models for the Evaluation and Selection of Software Packages*, Journal of Management Information Systems, vol. 6, n° 4, printemps 1990, p. 122-137.
- Anderson, E., Weitz, B.A.**, *Make or Buy Decisions: Vertical integration and Market Productivity*, Sloan Management Review, printemps 1986, p. 3-19.
- Applegate, L.M., Cash, J., Mills, D.Q.**, *Information Technology and tomorrow's manager*, Harvard Business Review, novembre-décembre 1988, traduit dans Harvard-L'expansion, automne 1990, p. 33-43.
- Applegate, L.M., Elam, J.J.**, *New Information Systems Leaders : A Changing Role in a Changing World*, MIS Quarterly, décembre 1992, p. 469-489.
- Ardoin, J-L., Michel, D., Schmidt, J.**, *Le contrôle de gestion*, Publi-Union, 2ème Ed., Paris, 1986.

- Arrow, K.J.**, *De la rationalité - de l'individu et des autres - dans un système économique*, Revue française d'économie, hiver 1987, p. 22-47.
- Avison, D.E., Wood-Harper, A.T.**, *Information Systems Development Research : An Exploration of Ideas in Practice*, The Computer Journal, vol. 34, n° 2, 1991, p. 98-112.
- Baiman, S.**, *Agency Research in Managerial Accounting : A Second Look*, Accounting Organizations and Society, vol. 15, n° 4, 1990, p. 341-371.
- Ballé, C.**, *Neutre mais pas innocente*, Revue Française de gestion, janvier-février 1979, p. 50-60.
- Barki, H., Hartwick, J.**, *Rethinking the Concept of User Involvement*, MIS Quarterly, mars 1989, p. 53-63.
- Barnard, C.I.**, *The functions of the executive*, Harvard University Press, 1938.
- Barreyre, P.Y.**, *L'impartition, politique pour une entreprise compétitive*, Librairie Hachette, Paris, 1968.
- Barzel, Y.**, *Measurement Cost and the Organization of Markets*, The Journal of Law and Economics, vol. 25, avril 1982, p.27-48.
- Benveniste, A.**, *Informatique et Systèmes industriels : la nouvelle donne*, Réalités Industrielles, mai 1991, p. 17-22.
- Birien, J. Y.**, *Histoire de l'informatique*, P.U.F., Paris, 1990.
- Blanc S.**, *Logiciels : développer ou acheter ?* 01 Références, n° 22, octobre 1992, p. 60-62.
- Bloomfield, B.P., Coombs, R.**, *Information technology, control and power : the centralisation and decentralisation debate revisited*, Journal of Management Studies, vol. 29, juillet 1992, p. 459-484.
- Boehm, B.W.**, *A spiral Model of Software Development and Enhancement*, Computer, mai 1988, p.61-72.
- Boehm, B.W., Gray, T.E., Seewaldt, T.**, *Prototyping Versus Specifying : A Multiproject Experiment*, IEEE Transactions on Software Engineering, vol. 10, n° 3, mai 1984, p. 78-90.
- Boynton, A.C, Zmud, R.W.**, *Information Technology Planning in the 1990's : Directions for Practice and Research*, MIS Quarterly, mars 1987, p. 59-71.
- Breton, P.**, *Histoire de l'informatique*, La Découverte, Paris, 1987.

- Brousseau, E.**, *L'approche néo-institutionnelle de l'économie des coûts de transaction*, Revue française d'économie, vol. 4, automne 1989, p. 122-166.
- Bryce, M., Bryce, T.**, *Make or buy software?*, Journal of systems management, août 1987, p. 6-11.
- Buchowicz, B.S.**, *A Process model of Make-vs-Buy Decision-making; The Case of Manufacturing Software*, IEEE Transactions on Engineering Management, vol. 38, n° 1, février 1991, p. 24-32.
- Buck-Lew, M.**, *To Outsource or Not ?*, International Journal of Information Management, n°12, 1992, p. 3-20.
- Cash, J.I., McFarlan, F.W., McKenney, J.L., Applegate, L.M.**, *Corporate Information Systems Management : Text and cases*, Richard D. Irwin, 1992.
- Causse, G., Chevalier, A., Hirsch, G.**, *Management financier - analyse, décision, contrôle*, Sirey, Paris, 1979.
- Causse, G., Lacrampe, S.**, *Analyse et contrôle des coûts - principes et systèmes*, Masson, Paris, 1981
- Chandler**, *The Visible Hand. The Managerial Revolution in American Business*, Harvard University Press, Cambridge, 1977.
- Charreaux, G., Couret, A., Joffre, P., Koenig, G., De Montmorillon, B.**, *De nouvelles théories pour gérer l'entreprise*, Economica, Paris, 1987.
- Charreaux, G.**, *La théorie positive de l'agence : une synthèse de la littérature*, in Charreaux et al., 1987, p. 21-55.
- Chokron, M., Reix, R.**, *Planification des systèmes d'information et stratégie de l'entreprise*, Revue Française de Gestion, janvier-février 1987, p. 12-21.
- Cigref**, *Evolution de la fonction informatique - la contribution du Cigref*, ref. 93830, octobre 1993.
- Coase, R.H.**, *The Nature of the Firm*, Economica N.S.G., nov. 1937, repris, traduit et commenté dans la Revue française d'économie, vol. II, 1, hiver 1987, p. 143-163.

- Cole, R.E.**, *Target information for competitive performance*, Harvard Business Review, mai-juin 1986, traduit dans Harvard L'Expansion, Les japonais et l'information, hiver 1986, p. 48-60.
- Collis, J.**, *A resource-based analysis of global competition*, Strategic Management Journal, vol. 12, 1991, p. 49-68.
- Commons, J.R.**, *Institutional economics : its place in political economy*, New York, Macmillan, 1934.
- Davis, G.B, Olson, M., Ajenstat, J., Peaucelle, J.L.**, *Systèmes d'information pour le management*, Editions G. Vermette, Economica, vol. 1 et 2, Paris, 1986.
- Dearden, J.**, *The Withering Away of the IS Organization*, Sloan Management Review, été 1987, p. 87-91.
- Delmond, M-H**, *L'utilisation du système d'information comme support de la décision et comme outil stratégique : tendances et facteurs applicatifs*, Mémoire de DEA, Université de Paris Val de Marne, 1990.
- Demski, J.S.**, *Information Analysis*, 2e ed., Addison Wesley, 1980.
- Desaintquentin, J.M., Sauteur, B.**, *Une nouvelle donne pour l'informatique*, Editions Synthèse informatique, Mareil sur Mauldre, 1993.
- Dumoulin, C.**, *Management des systèmes d'information*, Editions d'Organisation, Paris, 1986.
- Dupuy, Y., Rolland, G.**, *Manuel de contrôle de gestion*, Dunod, Paris, 1991.
- Dussauge, P., Ramanantsoa, B.**, *Technologie et stratégie d'entreprise*, McGraw Hill, Paris, 1987.
- Eisenhardt, K.M.**, *Agency Theory : An Assessment and Review*, Academy of Management Review, vol. 14, n° 1, 1989, p. 57-74.
- Eurostaf**, *Les S.S.I.I. européennes face à la concurrence internationale*, Collection "analyses de secteurs", 3^e trimestre 1989.
- Eymard-Duverney, F.**, *Conventions de qualité et formes de coordination*, Revue économique, n° 2, mars 1989, p. 329-359.
- Feeny, D.F., Edwards, B.R., Simpson, K.M.**, *Understanding the CEO/CIO Relationship*, MIS Quarterly, décembre 1992, p. 435-447.

- Ford, D., Farmer, D.,** *Make or Buy, A Key Strategic Issue*, Long Range Planning, vol. 19, n° 5, 1986, p. 54-62.
- Francalanci, C., Maggiolini, P.,** *Measuring the impact of investments in information technologies on business performance*, Politecnico de Milano, report n° 93.056, 1993.
- Franz, C.R., Robey, D.,** *Organizational Context, User Involvement, and the Usefulness of Information Systems*, Decision Sciences, juillet 1986, p. 329-356.
- Freeman, P.,** *Why Johnny Can't Analyse*, in Systems Analysis and Design, Ed. Cotterman W.W., Elsevier North Holland, New York, 1981, p. 321-329.
- Garette, B.,** *Les alliances entre firmes concurrentes : configurations et déterminants du management stratégique*, Thèse de doctorat, Ecole des Hautes Etudes Commerciales, Paris, 1991.
- Gershkoff, I.,** *The Make or Buy Game*, Datamation, 15/02/1990, p. 73-77.
- Gladden, G.R.,** *Stop the Life-Cycle, I Want to Get Off*, ACM SIGSOFT Software Engineering Notes, vol. 7, n° 2, avril 1982, p. 35-39.
- Glass, R.L., Vessey, I., Conger, S.A.,** *Software tasks : Intellectual or clerical ?*, Information & Management, n° 23, 1992, p. 183-191.
- Goldring, S.,** *La suprématie du code américain*, L'Informatique professionnelle, n° 113, avril 1993, p. 61-68.
- Grant, R. M.,** *The ressource-based theory of competitive advantage*, California Management Review, printemps 1991, p.114-135.
- Gremillion, L.L.,** *Improving Productivity with Applications Software Packages*, Business Horizons, mars-avril 1982, p. 51-54.
- Gremillion, L.L., Pyburn, P.,** *Breaking the System Development Bottleneck*, Harvard Business Review, p. 130-137, mars-avril 1983, p. 130-137.
- Guibert, P.,** *Vous n'avez jamais su développer du logiciel !*, L'informatique professionnelle, n°113, avril 1993, p. 47-52.
- Guimaraes, T., Saraph, J.V.,** *The role of prototyping in executive decision systems*, Information & Management, n° 21, 1991, p. 257-267.

- Hamel, G., Prahalad, C.K.**, *La stratégie à effet de levier*, Harvard-L'Expansion, été 1993, p. 43-54.
- Hansen, G.W.**, *Avoiding the Pitfalls of Installing Vertical Market Software*, Journal of Information Systems Management, printemps 1986, p. 43-48
- Hayek, F.**, *The Use of Knowledge in Society*, American Economic Review, vol. 35, 1945, p. 519-530
- Hayward, R.G.**, *Developping an Information Systems Strategy*, Long Range Planning, vol. 20, n° 2, 1987, p. 100-113.
- Henderson, J.C., Lee, S.**, *Managing I/S Design Teams: A Control Theories Perspective*, Management Science, juin 1992, p. 757-777.
- Hirshleifer, J., Riley, J.G.**, *The analysis of uncertainty and information - An expository survey*, Journal of Economic Literature, vol. 17, décembre 1979, p. 1375-1421.
- Holmstrom, B.R., Tirole, J.**, *The Theory of the Firm*, in Handbook of Industrial Organization, Ed. Schmalensee R., Willig, R.D., Elsevier Science Publishers, 1989, p. 61-133.
- Hopper, M.D.**, *Rattling sabre - New ways to compete on information*, Harvard Business Review, mai-juin 1990, p. 118-125.
- Huber, R.**, *Comment Continental Bank a sous-traité son informatique*, L'Expansion, n° 69, été 1993, p. 80-88.
- I/S Analyser**, *From application development to software engineering*, vol. 30, n° 7, juillet 1992.
- I/S Analyser**, *The costs and benefits of CASE*, vol. 31, n° 6, juin 1993.
- Ives, B., Olson, M.**, *Manager or Technician? The Nature of the Information Systems Job*, MIS Quarterly, vol. 5, n°4, décembre 1981, p. 49-63.
- Ives, B., Olson, M.**, *User Involvement and MIS Success: A Review of Research*, Management Science, vol. 30, n° 5, mai 1984, p. 586-603.
- Jarvenpaa, S.L., Ives, B.**, *Executive Involvement and Participation in the Management of Information Technology*, MIS Quarterly, juin 1991, p. 205-221.
- Jensen, M.C., Meckling, W.H.**, *Theory of the Firm : Managerial Behavior, Agency Costs and Ownership Structure*, Journal of Financial Economics, n° 3, 1976, p. 305-360.

- Jobber, D., Saunders, J., Gilding, B., Hooley, G., Hatton-Smooker, J.,** *Assessing the Value of a Quality Assurance Certificate for Software: an Exploratory Investigation*, MIS Quarterly, mars 1989, p. 19-31.
- Joffre, P.,** *L'économie des coûts de transaction*, in Joffre et al., 1985, p. 80-102.
- Joffre, P., Koenig, G.,** *Stratégie d'Entreprise : Antimanuel*, Economica, Paris, 1985.
- Kassouf, S.,** *Normative Decision Making*, Prentice-Hall Foundations of Administration Series, Englewood Cliffs, New Jersey, 1970.
- Ketler, K., Walstrom, J., Wagner J.L.,** *An overview of the outsourcing decision*, in Information Technology and Organizations : challenges of New Technologies, Ed. Mehdi Kosrowpow, Idea Group Publishing, London, 1994, p. 331-358.
- Lacity, M.C., Hirschheim, R.,** *Information Systems Outsourcing; Myths, Metaphors and Realities*, John Wiley and Sons, Wiley Series in Information Systems, 1993.
- Lacrampe, S., Macquin, A.,** *Les partenariats commerciaux, le cas de l'informatique*, Les cahiers de recherche, Groupe HEC, CR 459, 1993.
- Laigle,** *Un phénomène naturel et irréversible*, Le Monde Informatique, 07/01/1994, p. 18-20.
- Lambert, G.,** *Variables clés pour le transfert de technologie et le management de l'innovation*, Revue Française de Gestion, juin-juillet-août 1993, p. 49-72.
- Lamberterie, I.,** *Les contrats informatiques*, Jurisprudence Française, Librairie de la cour de cassation, 1983.
- Lamy,** *droit de l'informatique*, cf. Vivant et al. (1992).
- Launi, A.,** *Structured Methodology For Off-The-Shelf Software Implementation*, Journal of Systems Management, octobre 1991, p. 6-9.
- Lawrence, P., Lorsh J.W.,** *Adapter les structures de l'entreprise*, Les Editions d'Organisation, Paris, 1989, traduit de : Organization and Environment, Boston, Harvard University Press, 1986 (rééd.).
- Le Blanc, L.A., Jelassi, T.,** *An Empirical assessment of Choice Models for Software Evaluation and Selection*, Cahiers de Recherche de l'Insead, N° 91/24/TM, 1991, p. 1-24.
- Lederer, A.L., Sethi, V.,** *Meeting the Challenges of Information Systems Planning*, Long Range Planning, vol. 25, n° 2, 1992, p. 69-80.

- Le Gendre, O.**, *Une ardente obligation !*, L'Informatique Professionnelle n° 113, avril 1993, p. 5-10.
- Lesca, H.**, *Pour une direction des ressources d'information*, Revue Française de Gestion, septembre-octobre 1990, p. 40-44.
- Lieberman, M.B.**, *Determinants of vertical integration : an empirical test*, The Journal of Industrial Economics, vol.39, n° 5, septembre 1991, p.451-466.
- Llewellyn, K. N.**, *What price contract ? An essai in perspective*, Yale Law Journal, n° 40, 1931, p. 704-751.
- Loh, L., Venkatraman, N.**, *Determinants of Information Technology Outsourcing : A Cross-Sectional Analysis*, Journal of Management Information System, vol. 9, n° 1, été 1992, p.7-24.
- Lucas, H.C.**, *The Evolution of an Information Systems : From Key-Man to Every Person*, Sloan Management Review, vol. 19, n° 2, hiver 1978, p. 39-52.
- Lucas, H.C., Walton, E.J., Ginzberg, M.J.**, *Implementing Package Software*, MIS Quarterly, décembre 1988, p. 537-549.
- Lynch, R.K.**, *Nine Pitfalls in Implementing Packaged Applications Software*, Journal of Information Systems Management, printemps 1985, p. 88-92.
- Macneil, I.R.**, *The many futures of contracts*, Southern California Law Review, n° 47, 1974, p. 691-816.
- Malleret, V.**, *La sous-traitance : critères de décision et principes de gestion*, Institut de Management Public, R 3305 N, août 1985.
- Marjerin J., Ausset G.**, *Choix des Investissements, Présélection-Choix-Contrôle*, Ed. S.E.D.I.F.O.R. Paris, 1979.
- Martin, J., McClure, C.**, *Buying Software Off the Rack*, Harvard Business Review, novembre-décembre 1983, p. 32-60
- Martinet, A. CH.**, *Stratégie*, Vuibert gestion, Paris, 1983.
- Martinsons, M.G.**, *Outsourcing Information Systems : A Strategic Partnership with Risks*, Long Range Planning vol. 26, n° 3, 1993, p.18-25.
- Masten, S.E., Meehan, J.W., Snyder, E.A.**, *The costs of Organization*, The Journal of Law, Economics, & Organizations, vol. 7, n° 1, 1991, p. 1-25.

McFadden, F, Discenza, R., *Confronting the software crisis*, Business Horizons, novembre-décembre 1987, p. 68-73.

McFarlan, F.W., *Portfolio approach to Information Systems*, Harvard Business Review, septembre-octobre 1981, traduit dans Harvard-L'Expansion, Gérez vos systèmes d'information comme un portefeuille d'activités, printemps 1982, p. 37-48.

McFarlan, F.W, McKenney, J.L., Pyburn, P, *The Information Archipelago: Plotting a Course*, Harvard Business Review, janvier-février 1983, p. 145-156.

Micheaux, A., *Marketing : stratégies des bases de données*, Bancatque, n° 82, mai 1992, p. 264-270.

Miller, G.A., *The Magical Number Seven, Plus or Minus Two: Some Limits in Our Capability for Processing Information*, The Psychological Review, vol. 63, mars 1956, p.81-97.

Miller, G.A., *The Psychology of Communication*, New York, Basic Books, 1967.

Mintzberg, H., *The Nature of Managerial Work*, Harper & Row, N.Y., 1973.

Mintzberg, H., *Planning of the left side and managing on the right*, Harvard Business Review, juillet-août 1976, p. 49-58.

Mintzberg, H., *Le manager au quotidien*, Les Editions d'Organisation, 1984.

Mintzberg, H., *Le pouvoir dans les organisations*, Les éditions d'Organisation, Paris, 1986, traduit de Power In and Around Organizations, Prentice Hall Inc., Englewood Cliffs, New Jersey, 1983.

Morley, C., *Recherche sur le choix et la mise en pratique d'une méthode de développement de système d'information automatisé*, Thèse de doctorat, Ecole des Hautes Etudes Commerciales, 1991.

Mosakowski, E., *Organizational Boundaries and Economic Performance: an Empirical Study of Entrepreneurial Computer Firms*, Strategic Management Journal, vol 12, 1991, p. 115-133.

Moynihan, T., *What Chief Executives and Senior Managers Want From Their IT Departments*, MIS Quaterly, mars 1990, p. 15-25.

- Naumann, J.D., Jenkins, A.M.**, *Prototyping : The New Paradigm for systems development*, MIS Quarterly, septembre 1982, p. 29-49.
- Neumann, S., Ahituv, N., Zviran, M.**, *A measure for determining the strategic relevance of IS to the organization*, Information & Management, n° 22, 1992, p. 281-299.
- Niederman, F., Brancheau, J.C., Wetherbe, J.C.**, *Information Systems Management Issues for the 1990's*, MIS Quarterly, décembre 1991, p. 475-495.
- Nolan, R.L., Gibson, C.F.**, *Managing the Four Stages of EDP Growth*, Harvard Business Review, janvier-février 1974, p. 76-88.
- Nolan, R.L.**, *Business needs a new breed of EDP manager*, Harvard Business Review, mars-avril 1976, p. 123-133.
- Nolan, R.L.**, *Controlling the costs of data services*, Harvard Business Review, juillet-août 1977, p. 114-124.
- Nolan, R.L.**, *Managing the crises in data processing*, Harvard Business Review, mars-avril 1979, p. 115-126.
- Nutt, P.C.**, *Influence of decision styles on use of decision models*, Technological forecasting and social change, vol.14, n°1, juin 1979.
- OCDE**, *Les logiciels: l'émergence d'une industrie*, Rapport pour le Comité de la Politique de l'Information, de l'Informatique et des Communications, 1985.
- Orli, R.J.**, *Adopting Noncommercial Application Software*, Journal of Information Systems Management, hiver 1989, p. 55-60.
- Ouchi, W.**, *Markets, Bureaucracies and Clans*, Administrative Science Quarterly, vol. 25, mars 1980, p. 129-141.
- Perrow, C.**, *The Analysis of Goals in Complex Organizations*, American Sociological Review, 1961, p. 854-866.
- Pfeffer, J.**, *Power in organizations*, Pitman Publishing, Marshfield, Massachussets, 1981.
- Porter, M.E., Millar, V.E.**, *How Information gives you competitive Advantage*, Harvard Business Review, juillet-août 1985, p. 149-160.
- Prahalad, C.K., Hamel, G.**, *Les grands Groupes ne connaissent pas leur métier*, Harvard-L'Expansion, Hiver1990-91, p. 34-46.

- Quinn, J.B., Doorley, T.L., Paquette, P.C.**, *Technology in Services : Rethinking Strategic Focus*, Sloan Management Review, hiver 1990, p. 79-87.
- Quinn, J.B., Hilmer, F.G.**, *Strategic Outsourcing*, Sloan Management Review, Summer 1994, p. 43-55.
- Rands, T.**, *Software acquisition and changes in the software sector*, Management Research Papers, Templeton College, Oxford, MRP 90/11, 1990.
- Rands, T.**, *The key role of software make/buy decisions during the 1990's*, Management Research Papers, Templeton College, Oxford, MRP 91/5, 1991(a).
- Rands, T.**, *A Framework for managing software make/buy*, Management Research Papers, Templeton College, Oxford, MRP 91/8, 1991(b).
- Ravichandran, R., Ahmed, N.U.**, *Offshore System development*, Information & Management, n° 24, 1993, p. 33-40.
- Reix, R.**, *Marchés et organisations hiérarchisées à propos d'un ouvrage de O.E. Williamson*, Revue d'Economie Industrielle, n°8, 2ème trimestre, 1979(a), p. 110-113.
- Reix, R.**, *La flexibilité de l'entreprise*, Editions CUJAS, Paris, 1979(b).
- Reix, R.**, *L'impact organisationnel des nouvelles technologies de l'information*, Revue Française de Gestion, janvier-février 1990, p. 100-106.
- Robey, D., Markus, M.L.**, *Rituals in Information System design*, MIS Quarterly, mars 1984, p. 5-15.
- Robey, D., Taggart, W.M.**, *Measuring managers' minds : the assessment of style in human information processing*, Academy of Management Review, n° 6, 1981, p. 375-393.
- Romm, T., Pliskin, N., Weber, Y., Lee, A.S.**, *Identifying organizational culture clash in MIS implementation : when is it worth the effort ?*, Information & Management, n° 21, 1991, p. 99-109.
- Salzman, C.**, *La productivité des études n'est pas au rendez-vous*, L'informatique professionnelle, n° 113, avril 1993, p. 33-38.
- Sauviat**, *Les mutations du marché de l'expertise et du conseil*, Problèmes Economiques, n° 2238, 28 août 1991, p. 4-14.

- Selig, F.F.**, *Managing information technology in the nineties*, Information & Management, n°21, 1991.
- Sherer, S. A.**, *Purchasing software systems*, Information et Management, n° 24, 1993, p. 257-266.
- Sibley, E. H.**, *How to Select and Evaluate a DBMS*, Journal of Information Systems Management, printemps 1985, p. 40-49.
- Simon, H.A.**, *Administrative Behavior*, Macmillan, 2ème Ed., New York, (1ère Ed. 1947) 1958.
- Stephens, C.S., Ledbetter, W.N., Mitra, A., Ford, F.N.**, *Executive or functional manager? The Nature of the CIO's Job*, MIS Quarterly, décembre 1992, p. 449-467.
- Strassmann, P.A.**, *Managing the costs of information*, Harvard Business Review, septembre-octobre 1976, p. 133-142.
- Strassmann, P.A.**, *The business value of computers*, The Information Economics Press, New Canaan, 1990.
- Straub, D.W.**, *Validating Instruments in MIS Research*, MIS Quarterly, juin 1989, p.147-165.
- Sundem, G.L.**, *The Contributions of Information Economics : A Defense*, The Accounting Forum, vol. 52, décembre 1982, p. 1-13.
- Swanson, E.B.**, *Management Information Systems: Appreciation and Involvement*, Management Science, octobre 1974, p. 178-188.
- Swartout, W., Balzer, R.**, *On the Inevitable Intertwining of Specification and Implementation*, Communications of the ACM, vol. 25, n° 7, juillet 1982, p. 26-35.
- Tait, P., Vessey, I.**, *The Effect of User Involvement on System Success : A Contingency Approach*, Mis Quarterly, mars 1988, p. 91-107.
- Tallineau, Y.**, *Le responsable a des degrés de liberté*, L'informatique professionnelle, n° 113, avril 1993, p. 39-46.
- Tenhenhaus, M.**, *Méthodes Statistiques en Gestion*, Dunod, Paris, 1994.
- Toubol, F.**, *Le logiciel: analyse juridique*, Feduci L.G.D.J., Paris, 1986.
- Trenouth, J.**, *A Survey of Exploratory Software Development*, The Computer Journal, vol. 34, n° 2, 1991, p. 153-162.

- Venkatesan, R.**, *Strategic sourcing : to make or not to make*, Harvard Business Review, novembre-décembre 1992, p. 98-107.
- Venner, G.S.**, *Managing Applications as a Software Portfolio*, Journal of Information Systems Management, été 1988, p. 14-18.
- Vivant, M., Le Stanc, C., Rapp, L., Guibal, M., Costes, L.**, *Lamy droit de l'informatique*, LAMY S.A., Paris, 1992.
- Walker, M.**, *Agency Theory : A Falsificationist Perspective*, Accounting, Organizations and Society, vol. 14, n° 5/6, 1989, p. 433-453.
- Walker, G., Poppo, L.**, *Profit Centers, Single-Source Suppliers, and Transaction Costs*, Administrative Science Quarterly, mars 1991, p. 66-87.
- Walker, G., Weber, D.**, *A Transaction Cost Approach to Make-or-Buy Decisions*, Administrative Science Quarterly, septembre 1984, 373-391.
- Whang, S.**, *Contracting for Software Development*, Management Science, vol. 38, n° 3, mars 1992, p. 307-324.
- Ward, J.M.**, *Integrating Information Systems into Business Strategies*, Long Range Planning, vol. 20, n° 3, 1987, p. 19-29.
- Willcocks, L.**, *IT Evaluation : Managing the Catch 22*, European Management Journal, vol. 10, n°2, juin 1992, p. 220-229.
- Williamson, O.E.**, *Markets and Hierarchies: analysis and antitrust implications*, Free Press, New York, 1975.
- Williamson, O.E.**, *The Economic Institutions of Capitalism*, Free Press, New York, 1985.
- Williamson, O.E.**, *Transaction Cost Economics*, in Handbook of Industrial Organization, Ed. Schmalensee, R., Willig, R.D., vol.1, Elsevier Science Publishers, 1989, p. 135-182.
- Williamson, O.E.**, *Comparative Economic Organization : The Analysis of Discrete Structural Alternatives*, Administrative Science Quarterly, vol. 36, 1991(a), p. 269-296.
- Williamson, O.E.**, *Strategizing, Economizing, and Economic Organization*, Strategic Management Journal, vol. 12, 1991(b), p. 75-94.

Table des matières

INTRODUCTION	1
--------------------	---

PREMIERE PARTIE : LES DONNEES DU PROBLEME	7
--	----------

CHAPITRE 1 : ANALYSE DU MOUVEMENT D'EXTERNALISATION DES ACTIVITES INFORMATIQUES	9
--	----------

INTRODUCTION	10
--------------------	----

SECTION 1 - L'HISTOIRE DE LA GESTION DE LA RESSOURCE INFORMATIQUE DANS LES ENTREPRISES	12
---	----

I. Jusqu'au milieu des années 60 : les débuts de l'informatique.....	12
--	----

II. Jusqu'à la fin des années 70 : diffusion de l'informatique, organisation du marché du matériel, utilisation de la ressource interne pour le logiciel.....	15
---	----

III. Les années 80 : révolution de la micro-informatique, structuration de l'industrie du logiciel, extension de la gamme des services	19
--	----

Conclusion	25
------------------	----

SECTION 2 - LE DEBAT SUR L'EXTERNALISATION DE L'INFORMATIQUE.....	29
---	----

I. Définitions de l'externalisation	30
---	----

II. Externaliser quoi ?	31
-------------------------------	----

III. Externaliser pourquoi ? Les raisons évoquées en faveur de l'externalisation.....	32
---	----

3.1 L'externalisation annoncée par Dearden et l'analyse des stratégies de systèmes d'information de Parsons et Ward.....	33
--	----

3.2 Les arguments des partisans de l'externalisation : avant tout la baisse des coûts.....	38
--	----

III. Quelles sont les limites de l'externalisation ?	40
--	----

CONCLUSION.....	43
-----------------	----

CHAPITRE 2 : L'EXTERNALISATION DU DEVELOPPEMENT D'APPLICATIONS	45
INTRODUCTION	46
SECTION 1 - L'ACQUISITION DES PROGICIELS : UN ARBITRAGE DAVANTAGE CENTRE SUR LA NATURE DU PRODUIT QUE SUR L'ORIGINE DE LA RESSOURCE	48
I. A l'origine de la diffusion des progiciels : le "goulot d'étranglement du logiciel"	48
II. Avantages et inconvénients des progiciels	52
III. Le choix entre progiciel et développement spécifique interne : le modèle de Buchowicz.....	57
IV. Les travaux de Rands sur les choix intermédiaires	63
SECTION 2 - NOTRE CHAMP D'ETUDE : L'ARBITRAGE ENTRE RESSOURCE INTERNE ET EXTERNE POUR DES TRAVAUX SPECIFIQUES	65
I. La définition du champ	65
II. Les spécificités de la problématique d'externalisation appliquée au développement d'applications	67
2.1 La forme particulière des accords d'externalisation de l'activité de développement d'applications : le contrat par projet.....	67
2.2 Les arguments du débat sur l'externalisation du développement d'applications : une dissonance avec le discours général	70
CONCLUSION	74

SECONDE PARTIE : ETUDE ANALYTIQUE DES APPROCHES THEORIQUES. 75

CHAPITRE 3 : L'APPROCHE INDUSTRIELLE - ANALYSE DU PROCESSUS DE DEVELOPPEMENT SOUS L'ANGLE TECHNOLOGIQUE	79
INTRODUCTION	80
SECTION 1 - LE MODELE INDUSTRIEL : L'OPTIMISATION DU PROCESSUS ET DES COUTS DE PRODUCTION	81
I. Les avantages de la division du travail et de la spécialisation.....	81
1.1 La relation entre coût unitaire de production et volume de production.....	82
1.2 Les autres avantages de la spécialisation	84
1.3. Le calcul économique : analyse comparative des coûts	85

II. Les formes d'externalisation dans l'approche industrielle	87
2.1. L'étendue de l'externalisation	87
2.2. Les types de sous-traitance.....	89
III - Principaux éléments de l'approche industrielle	90
SECTION 2 - L'ETUDE DU PROCESSUS DE DEVELOPPEMENT VU SOUS L'ANGLE TECHNOLOGIQUE.....	92
I. L'évolution des modèles de développement	94
1.1 B.W. Boehm et l'histoire des modèles de développement.	94
1.2 Les nouveaux paradigmes de développement	99
1.3. Quel processus de développement alternatif au cycle de la cascade?	101
II. La gestion de la technologie informatique de développement par les entreprises	103
2.1 A quel stade de développement en sont les nouvelles technologies informatiques?	104
2.2 Un retard français dans la diffusion des nouvelles technologies?.....	106
2.3 La gestion de la diffusion des technologies informatiques au sein de l'entreprise.....	108
2.4 La nouvelle donne du choix des ressources technologiques	112
CHAPITRE 4 : L'APPROCHE STRATEGIQUE DE L'EXTERNALISATION.....	116
INTRODUCTION	117
SECTION 1 - L'APPROCHE STRATEGIQUE DE L'EXTERNALISATION.....	118
I. Les ressources de base de l'entreprise.....	118
II. Les avantages de l'externalisation des activités secondaires	120
III. Les risques de l'externalisation et les garanties nécessaires	122
SECTION 2 - L'APPROCHE STRATEGIQUE DE L'EXTERNALISATION APPLIQUEE AU DEVELOPPEMENT	125
I. La proposition de Rands d'un cadre d'analyse.....	125
SECTION 3 - LE DEVELOPPEMENT D'APPLICATIONS SPECIFIQUES PRESENTE-T-IL UN INTERET STRATEGIQUE FORT POUR L'ORGANISATION ?	127
I. La reconnaissance de l'impact stratégique des systèmes d'information : la thèse de Porter	127
II. Le rôle des nouveaux développements.....	130
III. La remise en cause de l'intérêt stratégique.....	134
3.1 les difficultés de mesure de l'impact stratégique	134
3.2 La remise en cause du principe de l'acquisition d'un avantage concurrentiel durable par le biais du système d'information.....	136
CONCLUSION.....	141

**CHAPITRE 5 : L'ETUDE DE LA RELATION CONTRACTUELLE - LES
APPORTS DE LA THEORIE DE L'AGENCE..... 143**

INTRODUCTION 144

SECTION 1 - LE LOGICIEL, OBJET CONTRACTUEL..... 146

I. L'identification du logiciel comme objet du droit..... 146

1.1 Le logiciel est un objet immatériel 147

1.2 La question de l'assimilation à une invention industrielle 147

1.3 La protection par droits d'auteur..... 149

II. Les dispositions du droit des contrats en matière de logiciel..... 150

2.1 Les différentes catégories de logiciel..... 151

2.2 Les différentes catégories de prestations 152

2.3 Les difficultés dans la formation des contrats..... 153

2.4 Les obligations des parties dans les contrats d'informatique..... 154

III. Synthèse des caractéristiques du produit et de la transaction 155

3.1 Caractéristiques du produit 155

3.2 Les caractéristiques de la transaction d'informatisation 156

SECTION 2 - L'APPORT DE LA THEORIE DE L'AGENCE A NOTRE TRAVAIL..... 160

I. Les hypothèses et concepts fondamentaux..... 161

II. Les risques comportementaux pré- et post-contractuels 164

2.1 La sélection adverse 164

2.2 Le Moral Hazard 166

III. Les coûts d'agence 167

IV. Le pilotage de la relation d'agence 168

CONCLUSION : APPORTS ET LIMITES DE LA THEORIE DE L'AGENCE POUR
NOTRE RECHERCHE..... 171

**CHAPITRE 6 : L'ETUDE DE LA RELATION CONTRACTUELLE - LES
APPORTS DE LA THEORIE DES COUTS DE TRANSACTION..... 174**

INTRODUCTION 175

SECTION 1 - LA THEORIE DES COUTS DE TRANSACTION 177

I. Définitions et rappel des hypothèses..... 178

1.1 Transactions, structures de gouvernance et coûts de transaction 178

1.2 Les hypothèses comportementales et l'influence de l'environnement 180

II. Les attributs des transactions.....	183
2.1 La spécificité des actifs.....	184
2.2 La difficulté d'évaluation du bien.....	186
III. Les caractéristiques des structures de gouvernance.....	188
3.1 Les capacités d'adaptation.....	188
3.2 Le régime d'administration des contrats (contract law).....	190
3.3 Le système d'incitations et de contrôle.....	191
3.4 Synthèse des caractéristiques des structures de gouvernance ; précision du lien avec la spécificité des actifs.....	192
 SECTION 2 - L'APPLICATION AU LOGICIEL.....	 194
I - Le développement en interne permet-il une bonne adaptation coopérative?.....	195
1.1 Les facteurs généraux susceptibles de limiter la coopération entre informaticiens et utilisateurs.....	195
1.2 La communication entre les informaticiens et les autres membres de l'organisation.....	201
1.3 La coopération avec les utilisateurs au sein des projets de développement.....	205
II - Attributs des transactions : spécificité des actifs et difficultés d'évaluation.....	211
2.1 Spécificité des actifs.....	211
2.2 Difficultés d'évaluation du bien.....	215
 CONCLUSION.....	 218
 CHAPITRE 7 : SYNTHÈSE DES APPROCHES THÉORIQUES - MODÈLE ET HYPOTHÈSES DE LA RECHERCHE EMPIRIQUE.....	 220
 SECTION 1 - LES PRINCIPAUX APPORTS DE L'ANALYSE THÉORIQUE.....	 222
I. Les enseignements de l'approche de la spécialisation industrielle.....	222
II. Les enseignements de l'approche stratégique (théorie de la ressource).....	223
III. Les enseignements de la théorie de l'agence.....	223
IV. Les enseignements de la théorie des coûts de transaction.....	224
 SECTION 2 - MODÈLE ET HYPOTHÈSES DE LA RECHERCHE EMPIRIQUE.....	 226

CHAPITRE 8 : RESULTATS DE LA RECHERCHE QUANTITATIVE 232

INTRODUCTION 233

SECTION 1 - DEFINITION DE L'ECHANTILLON ; PRESENTATION DES VARIABLES ET DES HYPOTHESES DETAILLEES 234

I. Critères de choix des entreprises et des projets 234

II. Les interlocuteurs rencontrés 237

III. Le choix des variables 238

3.1 Les variables descriptives des choix d'externalisation 239

3.2 Les variables correspondant aux déterminants des choix 241

3.3 Les données collectées 242

SECTION 2 - ANALYSE DESCRIPTIVE DES CHOIX D'EXTERNALISATION 244

I. Analyse exploratoire : Description des variables résultat (choix d'externalisation) 245

1.1 Fréquences et moyennes 245

1.2 Les solutions mixtes : combinaisons de modes différents au sein d'une même étape 248

1.3 Les variations entre les étapes 249

1.4 Etude des corrélations 252

II. Analyse en composantes principales des 9 variables résultat (choix d'externalisation) 256

2.1 Interprétation des axes 257

III. Elaboration d'une typologie de projets 266

3.1 L'interprétation des deux groupes de projets utilisant le forfait (groupes 1 et 2) 269

3.2 L'interprétation des deux groupes de projets utilisant la régie (groupes 3 et 5) 272

3.3 L'interprétation des deux groupes de projets "internes" (groupes 4 et 6) 275

3.4 Synthèse de la typologie 277

SECTION 3 - TEST DES HYPOTHESES 278

I. Analyse des variables explicatives : étude des corrélations 279

1.1 Corrélations entre les variables descriptives du projet 279

1.2 Corrélations entre les variables d'évaluation du poids des facteurs dans la décision 280

1.3 Corrélations entre les deux groupes de variables 281

II. Etude des corrélations entre les variables explicatives et les deux premiers axes principaux 282

2.1 Corrélations avec le premier axe principal (forfait/autres modes) 282

2.2 Corrélations avec le second axe principal (régie/interne) 284

2.3 Synthèse de l'étude des corrélations 285

III. Le lien entre les variables explicatives et la typologie : analyse de la variance.....	286
3.1 Résultats de l'analyse de variance.....	287
2.2 Interprétation des résultats.....	288
CONCLUSION.....	292
ANNEXE : QUESTIONNAIRE SUR LES PROJETS.....	295
CHAPITRE 9 : RESULTATS DE LA RECHERCHE QUALITATIVE.....	297
INTRODUCTION.....	298
SECTION 1 - LE LIEN AVEC LA STRATEGIE : LES OBJECTIFS ASSIGNES A L'ACTIVITE DE DEVELOPPEMENT.....	300
I. La perception du rôle et de la place de l'activité de développement.....	300
1.1 Une mission de support non directement stratégique.....	300
1.2 L'incertitude sur la pérennité de l'activité de développement au sein de l'organisation.....	302
II. L'objectif de synergie.....	307
2.1 Le développement de l'esprit de service.....	308
2.2 La recherche de la réactivité.....	309
2.3 L'impact sur les choix d'externalisation : le maintien en interne de l'étape d'analyse du besoin.....	312
III. L'objectif de maîtrise des coûts.....	316
3.1 Recherche de la flexibilité de la ressource.....	317
3.2 Le refus d'investissements supplémentaires.....	319
IV. Conclusion sur les aspects stratégiques.....	320
SECTION 2 - LES CONTRAINTES INDUSTRIELLES : VOLUMES ET FLUX DE PRODUCTION, TECHNOLOGIE, DIFFICULTES D'INTEGRATION.....	323
I. Facteur favorable à l'externalisation : l'évolution des volumes et flux de production.....	323
1.1 Une perception aiguë de l'irrégularité des flux de production.....	323
1.2 Les tentatives difficiles de reconversion du personnel interne.....	325
1.3 Les stratégies d'élargissement de la base de clientèle.....	327
1.4 Les stratégies d'évitement de capacité.....	328
II. Le poids de la technologie : la logique de spécialisation entre différents acteurs du marché.....	329
2.1 L'externalisation permet de mieux maîtriser une technologie de développement de plus en plus complexe et multiforme.....	329
2.2 L'investissement interne dans la technologie doit rester limité.....	331
III. Limite à l'externalisation : l'intégration des nouvelles applications aux systèmes existants.....	332
IV. Conclusion sur les aspects industriels.....	333

SECTION 3 - LE PROBLEME DU CONTROLE	336
I. Le pilotage de la relation interne	337
1.1 La sévérité du marché : le budget comme référence unique	338
1.2 L'ajustement coopératif	343
1.3 L'auto-contrôle de l'informatique interne	348
1.4 Le lien entre les modes de contrôle de l'informatique interne et les comportements d'externalisation.....	352
II. L'externalisation en régie : les évolutions d'un mode de pilotage intermédiaire	354
2.1 Un mode traditionnel "quasi-interne" : la régie longue durée	354
2.2 Le déclin de la régie longue durée	356
2.3 L'évolution vers un mode de fonctionnement plus proche du marché : la "régie forfaitée"	360
III. L'externalisation au forfait : un mode souhaité mais des problèmes de contrôle	362
3.1 Les situations d'utilisation du forfait	363
3.2 Les freins au développement du forfait	364
3.3 Le "bon usage" du forfait : les conditions de mise en oeuvre	368
3.4 Le forfait comme mode d'apprentissage interne	373
III. Conclusion sur le problème du contrôle	378
 SECTION 4 - REFLEXIONS SUR "MAITRISER L'EXTERNALISATION"	 382
4.1 Peut-on se passer d'un service informatique interne pour le développement d'applications ?	382
4.2 Le rôle de l'informatique interne dans la relation tripartite : entreprise - informatique interne - fournisseur extérieur	387
 ANNEXE : GUIDE D'ENTRETIEN SEMI-DIRECTIF	 392
 CONCLUSION GENERALE	 395
 REFERENCES BIBLIOGRAPHIQUES	 409
 TABLE DES MATIERES	 422
 TABLE DES ILLUSTRATIONS	 430

Table des illustrations

Liste des figures et tableaux

Chapitre 1

Figure 1.1.	Les prestations des SSII françaises en 1993 (source Syntec).....	22
Figure 1.2.	Evolution du taux de croissance des dépenses informatiques (source PAC)	23
Tableau 1.1.	Synthèse des évolutions de l'informatique	26
Tableau 1.2.	La classification des contrats de Facilities Management (Laigle, 1994).....	32
Tableau 1.3.	Les choix d'organisation des systèmes d'information : l'approche contingente de Ward	36

Chapitre 2

Figure 2.1.	Représentation courante du problème de conception des logiciels : métamorphoses d'un système au cours de son élaboration (source : OCDE, 1985).....	50
Figure 2.2.	L'arbitrage entre développement interne et acquisition de progiciel : le modèle de Buchowicz (1991).....	58
Figure 2.3.	Les choix intermédiaires entre tout faire et tout acheter (source : Rands, 1991b)	64
Figure 2.4.	Les caractéristiques des différents modes d'outsourcing en systèmes d'information (source : Loh et Venkatraman, 1992).....	68
Tableau 2.1.	L'approche industrielle du choix entre progiciel et développement interne	53
Tableau 2.2.	Limites de notre champ de recherche	67
Tableau 2.3.	Les projets candidats à l'outsourcing selon Buck-Lew (1992)	69

Chapitre 3

Figure 3.1.	L'organisation de la production dans l'approche industrielle.....	88
Figure 3.2.	Nos règles de description de l'externalisation	89
Figure 3.3.	Le cycle traditionnel de développement (source : Boehm, 1988).....	96
Figure 3.4.	Le nouveau processus de développement alternatif au cycle en cascade (d'après Benveniste, 1991)	101
Tableau 3.1.	Les "4 cas-types de sous-traitance" (Malleret 1985).....	90
Tableau 3.2.	Le développement des logiciels : une comparaison France / Etats-Unis (d'après S.Goldring, 1993).....	107
Tableau 3.3.	La diffusion des technologies informatiques dans l'organisation (d'après Cash et al., 1992).....	110
Tableau 3.4.	La nouvelle donne du choix des ressources technologiques.....	113
Tableau 3.5.	Synthèse de l'approche industrielle.....	115

Chapitre 4

Figure 4.1.	L'approche de la théorie de la ressource sur l'élaboration de la stratégie (source Grant, 1991).....	119
Figure 4.2.	Les avantages de l'externalisation des activités secondaires dans une perspective stratégique (d'après Quinn, Doorley et Paquette, 1990).....	121
Figure 4.3.	Le cadre d'analyse proposé par Rands (1991b) pour les choix de ressources en matière de logiciel.....	126
Figure 4.4.	L'impact des systèmes d'information sur la chaîne de la valeur (d'après Porter et Millar, 1985).....	128
Figure 4.5.	L'analyse sectorielle de l'impact des systèmes d'information : la matrice de Porter (source : Porter et Millar, 1985).....	129
Figure 4.6.	L'importance stratégique actuelle et future des systèmes d'information (d'après McFarlan et al., 1983).....	130
Figure 4.7.	Analyse stratégique du portefeuille d'applications (source Ward, 1987).....	132
Tableau 4.1.	Synthèse de l'approche stratégique.....	142

Chapitre 5

Tableau 5.1.	Guides contractuels du Syntec-Informatique et du CIGREF : les différentes prestations.....	152
Tableau 5.2.	Les obligations des parties dans les contrats d'informatique.....	155
Tableau 5.3.	Facteurs jouant sur les caractéristiques du produit ou de la transaction.....	159
Tableau 5.4.	Les hypothèses et le champ d'application de la théorie de l'agence (source : Eisenhardt, 1989).....	161
Tableau 5.5.	Le choix entre une rémunération contractuelle sur la base du comportement ou en fonction du résultat obtenu (d'après Eisenhardt, 1989).....	169
Tableau 5.6.	Synthèse de l'approche de la théorie de l'agence.....	115

Chapitre 6

Figure 6.1.	Schéma des imperfections organisationnelles (source : Williamson, 1975).....	180
Figure 6.2.	Efficiences comparées de la hiérarchie et du marché en fonction du degré de spécificité des actifs (source : Williamson, 1985).....	185
Figure 6.3.	Profil et occupations des responsables de systèmes d'information : principaux résultats de l'étude de Applegate et Elam (1992).....	203
Figure 6.4.	Premiers modèles de participation des utilisateurs, (d'après Swanson, 1974, résultats contestés par Ives et Olson, 1984).....	206
Figure 6.5.	Second groupe de modèles de participation des utilisateurs (d'après Jarvenpaa et Ives, 1991).....	209

Tableau 6.1.	L'analyse de Barzel (1982) des problèmes d'évaluation des biens.....	187
Tableau 6.2.	Conditions d'efficacité des deux formes d'adaptation (d'après Williamson, 1991a).....	189
Tableau 6.3.	Correspondance entre structures de gouvernance et régimes d'administration des contrats (d'après Williamson, 1991a)	191
Tableau 6.4.	Caractéristiques des structures de gouvernance (source : Williamson, 1991a).....	193
Tableau 6.5.	Les deux visions des tâches de développement	199
Tableau 6.6.	Caractéristiques des différentes étapes du processus de développement et hypothèses de choix d'organisation dans le modèle des coûts de transaction.....	214
Tableau 6.7.	Synthèse de l'approche coûts de transaction	219

Chapitre 7

Tableau 7.1.	Synthèse des enseignements des différentes approches	225
Tableau 7.2.	Modèle global et hypothèses de la recherche empirique.....	228

Chapitre 8

Figure 8.1.	Evolution des choix d'externalisation entre les différentes étapes, sur les 60 projets.....	250
Figure 8.2.	Taux d'utilisation moyen des trois modes d'organisation dans les différentes étapes du processus	251
Figure 8.3.	Les corrélations entre l'utilisation d'une même solution tout au long du projet.....	253
Figure 8.4.	Les corrélations interne / régie et interne / forfait.....	253
Figure 8.5.	Les corrélations régie / autres modes et forfait / autres modes	253
Figure 8.6.	Cercle des corrélations sur le premier plan principal.....	259
Figure 8.7.	Représentation des projets sur le premier plan principal.....	260
Figure 8.8.	Cercle des corrélations sur le second plan principal (axe 1 ; axe 3)	262
Figure 8.9.	Représentation des projets sur le second plan principal	263
Figure 8.10.	Cercle des corrélations sur le troisième plan principal (axe 2 ; axe 3).....	264
Figure 8.11.	Représentation des projets sur le troisième plan principal.....	265
Figure 8.12.	Représentation de la typologie sur le premier plan principal.....	268
Figure 8.13.	Représentation de la typologie sur le second plan principal (axe 1 ; axe 3)	269
Figure 8.14.	Représentation de la typologie sur le troisième plan principal (axe 2; axe 3)	273
Tableau 8.1.	Profil des personnes rencontrées	238
Tableau 8.2.	Variables descriptives des choix d'externalisation pour chaque projet.....	240
Tableau 8.3.	Les solutions mixtes : combinaison de différents modes d'organisation au sein d'une même étape de projet	248
Tableau 8.4.	Matrice des corrélations entre les choix d'externalisation	252
Tableau 8.5.	Corrélations entre les variables d'origine et les composantes principales	258
Tableau 8.6.	Les deux logiques d'externalisation au forfait : détail des projets	270
Tableau 8.7.	Les deux groupes de projets en régie : détail des projets	274

Tableau 8.8.	Les deux groupes de projets gérés en interne.....	276
Tableau 8.9.	Synthèse de la typologie.....	277
Tableau 8.10.	Corrélations entre le premier axe principal et les variables explicatives.....	283
Tableau 8.11.	Corrélations entre le second axe principal et les variables explicatives.....	284
Tableau 8.12.	Synthèse des résultats de l'étude des corrélations.....	285
Tableau 8.13.	Résultats de l'analyse de variance.....	287

Chapitre 9

Figure 9.1	L'évolution de l'organigramme des services études.....	314
Figure 9.2.	La double relation de contrôle.....	387

