



HAL
open science

Le pilotage des représentations au risque de l'informatique

Thierry Bertrand

► **To cite this version:**

Thierry Bertrand. Le pilotage des représentations au risque de l'informatique. 21e congrès de l'AFC, Association française de comptabilité, May 2000, Angers, France. pp.CD-Rom. halshs-00587427

HAL Id: halshs-00587427

<https://shs.hal.science/halshs-00587427>

Submitted on 20 Apr 2011

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.

Le pilotage des représentations au risque de l'informatique

Thierry Bertrand¹

Résumé :

Dans le cadre d'une autonomie accrue, le pilotage des représentations des acteurs se substitue au contrôle de leurs actions. Cet article s'intéresse d'une part à l'impact du système d'information sur ce pilotage et d'autre part au risque que ce dernier n'échappe aux managers.

Mots clés :

Autonomie, représentations, pilotage, système d'information.

Abstract :

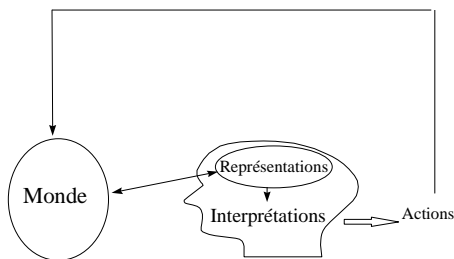
In the scope of empowerment, the management of the actors interpretations substitutes for the control of their actions. This paper focuses on the computer system role regarding this substitution. Therefore, it handles the resultant risk that this influence arises from the computer experts and no longer from the managers

Keywords :

Empowerment, management, information system

¹ Professeur agrégé du secondaire, en thèse sous la direction du professeur Mevellec au Laboratoire de Recherche en Sciences de Gestion de l'Université de Nantes. Enseignant au département informatique de la faculté des Sciences et Techniques : 2 rue de la Houssinière. BP 92208. 44322 Nantes cedex 03. Tel : (33)0251125830. Télécopie : (33)0251125812. Courrier électronique : Thierry.Bertrand@irin.univ-nantes.fr.

Dans l'expression « pilotage des représentations », empruntée à Lorino [1995], le terme de représentation renvoie à celui de connaissance. Pour M. Minsky [1988], les deux notions sont mêmes pratiquement substituables¹. Le choix du mot représentation enrichit néanmoins celui de connaissance d'une double métaphore [encyclopédie UNIVERSALIS]. N'ayant pas de contact direct avec le monde qui nous environne, la représentation, à l'instar d'un diplomate, médiatise notre relation au monde. A ce titre, la représentation rend présent le monde en nous. L'autre versant de la métaphore fait référence à la représentation théâtrale qui constitue une mise en scène reflétant autant la vision de son auteur que le monde lui-même. De ce point de vue, la représentation n'est pas simplement une somme de connaissances, mais une interprétation de ces connaissances, propre à chaque individu. Le terme de pilotage fait référence à la tentative de peser sur les représentations d'autrui. Il se place dans le cadre de l'autonomie accrue accordée aux membres d'une organisation. Cette autonomie accordée à un individu revient à le reconnaître en tant qu'entité cognitive capable de relier ses actions à ses propres interprétations (figure 1).



1 L'acteur : une entité cognitive reconnue

L'acteur utilise ses propres représentations pour collecter des données en provenance du monde puis pour les interpréter. Ses actions sur le monde sont alors le fruit de ces interprétations et seront à leur tour objet d'interprétations. De simple exécutant d'actions pensées par d'autres, l'individu passe alors au statut d'acteur. Dès lors, « l'action sur les actions » ne peut plus s'exercer directement mais seulement par l'intermédiaire du pilotage des représentations.

Comme le signale Lorino [1996] ce changement de statut ne se contente pas d'introduire du « flou et du mou » dans l'organisation mais constitue, plus fondamentalement, un changement de paradigme qui bouleverse complètement l'activité cognitive qui se développe en son sein. Non seulement l'intégration des représentations propres aux acteurs constitue un élargissement de la base cognitive dans l'organisation, mais elle en modifie aussi la structure. A cette transformation des représentations à l'œuvre dans l'organisation répond celle du système d'information qui les alimente en données. Or comme l'a montré Piaget [1970], données et représentations sont intimement liées. Les représentations d'un individu orientent son regard sur le monde et le conduisent à filtrer les données qui lui parviennent. En retour, l'assimilation de ces données aux représentations, voire l'accommodation de celles-ci aux données perçues, modifient les représentations.

¹ Marvin Minsky [1988] utilise en fait le terme de modèle : « Jacques a des connaissances sur A signifie qu'il y a un modèle de A dans la tête de Jacques ». « La société de l'esprit » édition française Inter-Editions p586

De ce fait, la problématique du pilotage englobe celle de l'impact du système d'information sur les représentations.

Pour traiter de cette problématique nous tenterons d'approfondir les implications du changement de paradigme organisationnel sur l'activité cognitive au sein de l'organisation. Cette exploration nous amènera à insister sur la place privilégiée du système informatique servant de support au système d'information.

Ce constat nous conduira alors à nous interroger sur la compatibilité des méthodologies de conception de ces systèmes avec leur pilotage par les managers. Cette mise en question amènera à mettre en lumière l'intérêt d'un langage de modélisation commun aux managers et aux informaticiens.

Nous explorerons alors les perspectives qui peuvent s'ouvrir en ce domaine.

1. Changement de paradigme et activité cognitive au sein des organisations.

H Mintzberg [1998] définit l'organisation comme : « un ensemble de personnes entreprenant une action collective à la poursuite de la réalisation d'une action commune ». La problématique de l'action commune recouvre celle de son articulation avec l'activité cognitive propre aux individus qui composent l'organisation. Nous nous placerons d'abord dans un cadre général pour préciser les différents niveaux auxquels se pose cette problématique, puis nous expliciterons comment le paradigme classique amène à la traiter et les limites de cette approche. L'autonomie accrue des acteurs apparaissant comme une tentative de réponse à ces limites, nous serons amenés à envisager ses implications en terme de pilotage.

1.1. Action commune et activité cognitive : les différents niveaux de la problématique.

La problématique peut être éclairée par simple duplication du schéma de base illustrant le fonctionnement d'une entité cognitive (figure 2).

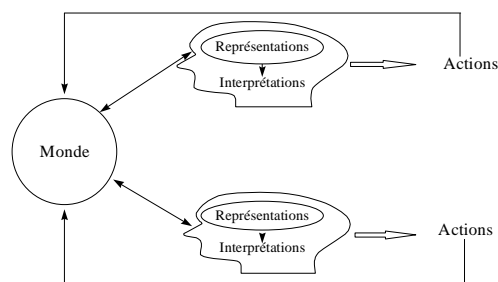


Figure 2 : Activité cognitive et action commune

La problématique de l'action commune peut être envisagée sous l'angle de sa pertinence, de son efficacité et de son efficacité.

L'organisation étant un système finalisé [Tabatoni et Jarniou], la pertinence des actions doit être mesurée par rapport aux finalités de l'organisation. Ces actions découlant des représentations propres aux individus, rien ne permet de présumer, a priori, leur adéquation aux objectifs de l'organisation.

L'efficacité est conditionnée par l'insertion des actions dans un cadre collectif, ce qui suppose leur coordination. Les représentations propres aux individus les amènent à des perceptions puis des interprétations différentes de l'organisation et du monde qui l'environne. Cette hétérogénéité ne permet nullement de présager l'intégration spontanée de ces actions individuelles dans une action collective.

Contrairement à l'activité cognitive, seule l'action est directement monnayable par l'organisation. Dès lors, l'efficacité suppose que l'investissement cognitif permettant les actions reste modéré.

1.2. Résolution de la problématique dans le cadre du paradigme classique.

1.2.1. Architecture générale

Le paradigme classique correspondant à une vision mécaniste de l'organisation repose sur une stricte séparation entre les phases de conception et d'exécution des actions (figure 3).

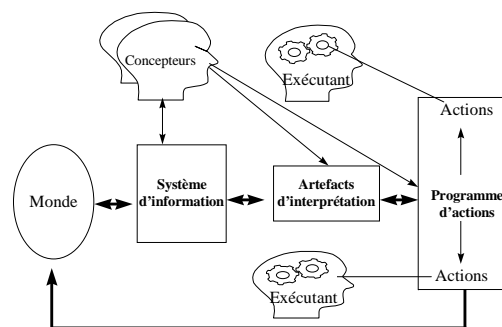


Figure 3 : L'architecture mécaniste

La phase de conception permet de régler à l'avance les problèmes de finalisation et de coordination des actions en les faisant reposer sur des artefacts d'interprétation et des programmes d'actions. L'utilisation répétitive de ces derniers permet de résoudre le problème d'efficacité en amortissant l'investissement cognitif réalisé lors de la phase de conception. On retrouve dans cette approche le rôle privilégié joué par la standardisation des procédés décrite par Mintzberg [1982].

1.2.2. Rôles du système d'information et caractéristiques du support informatique

Dans cette architecture, le système d'information remplit trois rôles essentiels : éclairer les concepteurs sur l'organisation et ses interactions avec l'environnement, aider à l'exécution des programmes d'actions et faciliter le contrôle de la conformité de cette exécution à ces programmes. Les problèmes d'exécution et de contrôle étant particulièrement bien structurés, l'informatisation s'est d'abord développée sous ces deux aspects.

Elle a permis de gagner en productivité en accompagnant pas à pas les phases d'exécution par apports et transferts de données. Les applications informatiques issues de cette démarche sont alors totalement dédiées aux programmes d'actions pour lesquels ils ont été développés. De ce fait, ils en épousent la structure et le niveau très fin de granularité (conception au niveau des tâches à exécuter).

La démarche analytique adoptée pour concevoir les organisations de type mécaniste aboutit à un faible couplage entre les entités de l'organisation, ce qui a pour corollaire la faible

densité de leurs échanges informationnels. En conséquence, les liens entre les applications informatiques peuvent se borner à des transferts de données à la fois limités et prédéfinis. De ce fait, les organisations héritent aujourd'hui de myriades d'applications informatiques hétérogènes dédiées à des besoins spécifiques. Certes, une double évolution, à la fois organisationnelle et technologique, a conduit à multiplier les connexions, donnant ainsi une apparence de système à ce qui n'était qu'une juxtaposition d'applications largement indépendantes les unes des autres. En réalité, les liens ainsi tissés après coup apparentent la plupart des ensembles informatiques davantage à des patchworks d'éléments disparates qu'à de véritables systèmes. C'est probablement l'explication principale du succès commercial impressionnant rencontré par les solutions intégrées proposées par des sociétés comme SAP ou ORACLE.

Dans les programmes d'actions, le problème du sens est totalement pris en charge lors de la conception. Il en va de même en ce qui concerne le développement des applications informatiques qui leur sont dédiées. En tant qu'exécutant, l'utilisateur se contente d'un nombre limité de signes qu'il n'a pas à interpréter. L'informaticien peut ainsi évacuer le problème du sens que l'utilisateur pourra attribuer à ce signe. Par contre, il doit encapsuler le sens du programme d'actions dans son propre programme informatique. Les applications informatiques s'assimilent alors à des chaînes de computation et de transfert de signes dont le sens a été préalablement et définitivement encapsulé dans le programme (figure 4).

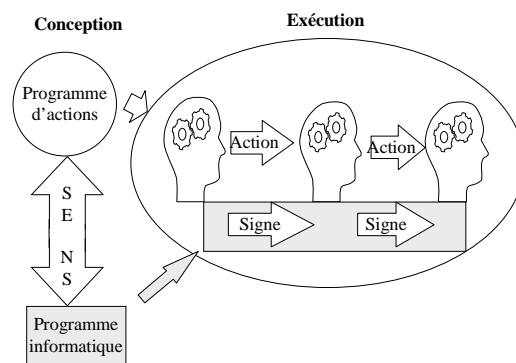


Figure 4. Le problème du sens dans les applications supports d'exécution

Les caractéristiques des systèmes informatiques dont nous héritons apparaissent comme les répliques des traits saillants des organisations mécanistes au sein, et pour lesquelles, ils ont été conçus :

1. éléments à base de chaînes de traitement à la fois totalement finalisées, très détaillées et fortement intégrées,
2. connections entre applications limitées à des transferts de données,
3. encapsulation du sens lors de la programmation.

1.2.3. Les limites du paradigme classique

Le paradigme mécaniste n'aurait pu s'assurer une position dominante si ce type d'architecture n'avait fait preuve, durant des décennies, d'une remarquable efficacité. Pour autant, ces succès ne doivent pas masquer les conditions sur lesquelles ils reposent.

La création d'artefacts suppose, en amont, la capacité des concepteurs à modéliser l'organisation, son environnement et les relations qui les relient. Cela n'est possible qu'en deçà d'un certain niveau de complexité. Tant que les concepteurs ne sont confrontés qu'à une complication croissante des problèmes à résoudre, la décomposition analytique de ces derniers

permet d'y faire face. Mais lorsque les imbrications et les interactions prennent un caractère récursif la complexité ne peut plus être traitée globalement au niveau d'un modèle détaillé. La simplicité du contexte apparaît dès lors comme une hypothèse implicite à l'adoption de ce type d'architecture.

La séparation des tâches de conception et d'exécution réduit nécessairement la réactivité de l'organisation face aux changements qui l'affectent. Tant que l'environnement reste suffisamment stable, ou du moins prévisible, cette inertie cause peu de dommages, au delà elle devient rédhibitoire.

Lorsque leur environnement dépasse ces seuils, les organisations sont amenées d'une part à répondre à la complexité croissante en élargissant leur base cognitive aux représentations des acteurs et d'autre part à les autoriser à concevoir eux mêmes leurs propres actions pour accroître la réactivité. Mais l'autonomie accrue ainsi accordée aux acteurs porte en germe des risques de perte de cohérence.

1.3. Les conditions de déploiement de l'autonomie accrue des acteurs.

1.3.1. La problématique du pilotage.

Le simple retour au schéma initial (figure 2) laisserait entiers les problèmes de pertinence, d'efficacité et d'efficience.

La pertinence des actions par rapport aux finalités de l'organisation suppose que les acteurs intègrent ces finalités dans leur propres représentations. Pour pouvoir inscrire leur propres actions locales dans le cadre global de l'organisation, ils doivent disposer à la fois d'une compréhension de leur place dans l'organisation et d'une vision stratégique de leur propre action. Comme l'écrit Lemoigne [1996] : « il faut développer la téléologie de l'action collective »

L'efficacité repose sur la capacité des acteurs à s'inscrire dans une action collective en gérant leurs interdépendances. Celles-ci ne sont pas nouvelles, mais elle revêtent un caractère inédit sous l'aspect cognitif. Tant qu'elles sont résolues par des normes et programmes, les représentations des exécutants n'interfèrent pas les unes sur les autres. Prisonniers des règles, les individus restent libres les uns par rapport aux autres. Au contraire, les acteurs libérés des procédures deviennent dépendants des représentations d'autrui. A l'articulation local-global de leurs propres actions dans un cadre stratégique, les acteurs doivent ajouter les articulations local-local de leurs interactions. Ainsi, la dimension collective des actions de chacun, antérieurement prise en charge en amont par la standardisation, trouve dorénavant sa source dans l'ajustement mutuel [Mintzberg 82]. Mais ce dernier ne peut se décliner en terme d'actions que s'il concerne d'abord les représentations : « l'organisation est une entité visant à dépasser l'écart existant entre connaissance et action en chaque individu et entre individus. » [Jacot et Micaelli 1996].

L'efficience suppose à la fois un partage des ressources entre acteurs et une prise en compte de leurs représentations au niveau organisationnel : « L'organisation est un artefact des modes individuels de représentation de l'action. L'organisation est une investigation réflexive » [Argyris et Schön cité par Lorino dans Ecosip 1996].

L'autonomie accrue des acteurs constitue de ce fait une mutation vers des organisations de type « réseau distribué ». L'ordre y émerge essentiellement de la densité des liens entre les noeuds du réseau. Cet ordre ne passe plus par la recherche et le maintien d'un état idéal

stationnaire mais par celle d'une dynamique d'ajustement permanent permettant un équilibre sans cesse détruit et donc toujours à retrouver. A l'image du tourbillon capable de se recréer en permanence derrière une pile de pont, l'ordre n'est plus le fait d'un état statique quasi minéral mais celui d'une dynamique permanente ordre-désordre [Morin, 1977]. Dès lors que la cohérence prend un caractère émergent, les managers perdent le pouvoir de la dessiner a priori. Pourtant, sauf à s'en remettre au hasard, c'est à eux qu'il revient de créer les conditions de son émergence. Cessant d'être les constructeurs de l'ordre lui-même, ils deviennent responsables du contexte dans lequel l'organisation pourra trouver sa cohésion. Ne pouvant intervenir directement au niveau des actions, ils ne peuvent agir que par l'intermédiaire du pilotage des représentations [Lorino, 95] (figure 5).

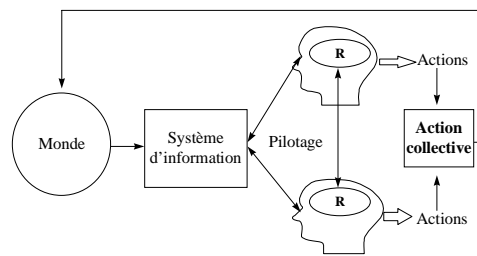


Figure 5 : La problématique du pilotage

Le pilotage consiste à mettre à disposition des acteurs les ressources cognitives qui leurs sont nécessaires. Cela amène notamment à leur fournir les vues de l'organisation nécessaires à l'enrichissement de leurs représentations, à leur proposer des représentations de références leur permettant d'ancrer les leurs dans une perspective collective et à mettre à leur disposition les support de communication indispensables à l'ajustement mutuel. Sur chacun de ces aspects, le système d'information, et son support informatique, jouent un rôle prépondérant.

1.3.2. Place du système d'information et caractéristiques du système informatique

De support à l'exécution et au contrôle le système d'information devient une aide à l'interprétation et à la communication.

L'autonomie n'aurait aucune réalité si elle ne se traduisait par une certaine latitude laissée aux acteurs quant à la définition et à l'agencement des tâches nécessaires à la réalisation des activités dont ils sont responsables. Sur le plan organisationnel, les activités deviennent ainsi des boîtes noires dont l'organisation interne relève du domaine privé des acteurs. Il en résulte que la coordination ne peut plus se concevoir au niveau des tâches mais à celui des activités. Les applications informatiques développées en support des programmes d'actions présentent alors un niveau de granularité beaucoup trop fin. Sauf à se priver de ce support, les acteurs sont alors incités à rester dans le cadre des programmes d'actions prédéfinis. Leurs représentations se nourrissant de leurs actions, c'est toute l'évolution du système cognitif qui s'en trouve partiellement bloquée. La double articulation interprétation-action que suggère le néologisme « Interprétation », proposé par Lemoigne, se trouve alors démantelée.

Contrairement aux exécutants, les acteurs ne sont pas de simples manipulateurs de signes mais des entités cognitives chargées d'interpréter. C'est au système d'information qu'il revient de leur apporter les vues de l'organisation et de son environnement nécessaires aux articulations global-local et local-local de leurs actions. La quantité de données à fournir pour

éclairer le contexte de décision d'un acteur est sans commune mesure avec ce qui est nécessaire à une simple exécution. Néanmoins, sur le plan strictement quantitatif, l'informatique est aujourd'hui parfaitement capable de relever le défis. Pourtant, mêmes si les entrepôts de données, voire les bases de données, sont bien susceptibles de mettre à disposition de l'acteur la matière première informationnelle dont il a besoin, celui-ci ne sera pas en mesure de la traiter dans un temps raisonnable. Au préalable, ces données doivent être mises en forme, (« in-formées »), pour constituer des vues éclairant le contexte de l'acteur. Or in-former les données c'est déjà les interpréter. En retour, ces interprétations pèsent fortement sur celles des acteurs. « Piloter, c'est construire des méta-interprétations permettant aux acteurs de construire leurs propres interprétations » [Lorino 1996]. Le pilotage des représentations passe donc par celui du système d'information qui les nourrit.

L'interprétation d'un acteur constitue une donnée que d'autres auront à interpréter à leur tour. Le sens global émerge donc, par strates successives, d'une chaîne où chaque acteur apporte sa contribution (figure 6).

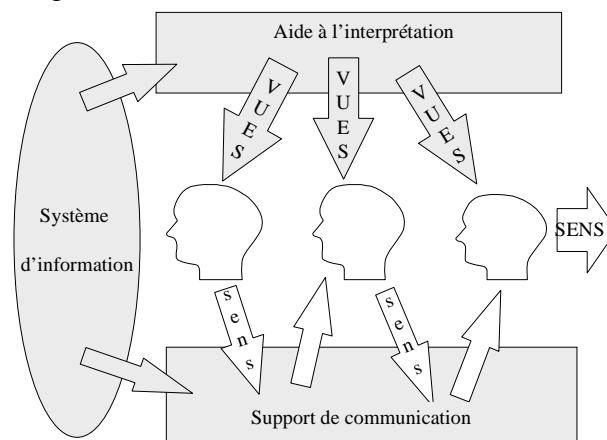


Figure 6. Processus de construction de sens

L'image de la chaîne relève néanmoins de la simplification dans la mesure où d'une part les imbrications ne sont pas linéaires d'un acteur au suivant mais multiples et multi-directionnelles et où d'autre part les interprétations ne sont jamais définitives mais toujours en reconstruction. Le processus d'attribution de sens doit être vu comme un système dont la production est réinjectée dans le système lui-même. Dès lors, le système d'information, et son support informatique, doivent permettre la densification des liaisons informationnelles nécessaires au développement de ce processus. Largement conçues pour assurer de simples transferts de données, les applications informatiques dont nous héritons se montrent déficientes lorsqu'il s'agit de véhiculer du sens.

Pour Lorino [1996], la communication entre acteurs ne peut se développer efficacement que si elle repose sur le partage de représentations qu'il qualifie de « publiques ». En tant que références communes, ces représentations « partagées » constituent un des ciments de l'organisation. Derrière l'architecture du système informatique se profile nécessairement le modèle sur lequel il est bâti. Le système lui-même étant partagé par l'ensemble des membres de l'organisation, le modèle qui lui est sous-jacent acquière la qualité de représentation publique. Celle-ci est d'autant plus puissante qu'elle est véhiculée par l'utilisation permanente de l'outil informatique qu'elle supporte. Cela lui confère un caractère concret validé par le vécu quotidien, ce dont ne dispose pas d'autres représentations publiques comme les

organigrammes. Ainsi, la vision véhiculée par un système « patchwork » renforce-t-elle l'image d'une organisation cloisonnée, ce qui rentre en contradiction avec celle du réseau que suppose l'autonomie accrue des acteurs. A contrario, dans le cadre d'un changement organisationnel, la modification de l'architecture du système informatique constitue une référence tangible aidant les acteurs à reconstruire leurs représentations. Il en va des systèmes informatiques comme des systèmes de calculs de coûts à propos desquels Mevellec [1996] écrit : « L'appréciation d'un système de coûts repose sur l'analyse de la cohérence entre le modèle implicite de l'entreprise qu'il véhicule et le modèle réel de fonctionnement de celle-ci. ».

La problématique de l'adaptation du système d'information dépasse donc largement celle de l'alimentation des acteurs en données pertinentes pour s'élargir à celle du pilotage des représentations. De par le rôle primordial que joue le système d'information en ce domaine, les managers ne peuvent prétendre piloter efficacement les représentations s'ils ne maîtrisent pas l'évolution du système d'information et de son support informatique. Dès lors, il apparaît important de questionner les méthodologies de conception des systèmes informatiques actuellement en vigueur pour évaluer leur compatibilité avec leur pilotage par les managers.

2. Méthode de conception des systèmes informatiques.

En nous fondant sur une présentation de la démarche classique de conception des systèmes informatiques, nous étudierons sa pertinence suivant les objectifs recherchés. Cette analyse nous amènera à souligner les risques que porte en germe cette démarche quant au pilotage des représentations.

2.1. Caractéristiques de la démarche classique.

S'il existe une grande diversité de démarches de conception et de développement des applications informatiques (méthode dites « en V », prototypage, etc), dès qu'il s'agit de concevoir un système complet, l'ensemble des méthodes revient, à quelques variantes près, à une démarche en quatre phases (figure 7). Une phase initiale de spécification permet de collecter les besoins auxquels devra répondre le futur système. Cette étape revêt une importance cruciale puisqu'elle doit assurer l'expression des besoins des utilisateurs et leur compréhension par les informaticiens. La phase d'analyse vise à traduire ces besoins en terme de fonctionnalités que devra remplir le futur système. S'appuyant sur cette analyse, la phase de conception consiste à bâtir le modèle du système qui prendra forme concrète lors du développement.

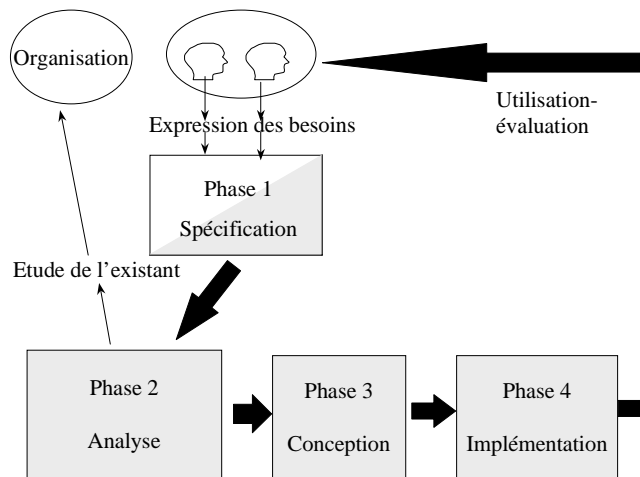


Figure 7 : Processus classique d'élaboration d'une application informatique

Les utilisateurs et les donneurs d'ordres, parmi lesquels se situent les managers, se positionnent à la fois à la source (spécification des besoins) et à l'embouchure (évaluation du système) du processus. De ce fait, et même si les phases grisées sont prises en charges par les informaticiens, les managers semblent contrôler le processus. Pourtant, une analyse plus fine amène à tempérer cette impression première. Partant de besoins abstraits, le processus doit aboutir à des réponses concrètes sous forme de lignes de code informatique. Ce faisant, il présente deux caractéristiques majeures.

1. Chacune des phases constitue une traduction de la précédente. Par essence, toute traduction repose sur une interprétation, en l'occurrence il s'agit ici de celle des informaticiens. Normalement, chacune de ces traductions fait l'objet d'une validation de la part du donneur d'ordre. Ce contrôle est néanmoins très partiel. D'une part, il devient très difficile lorsque la traduction est exprimée dans un langage ésotérique difficilement accessible aux non informaticiens. Ce qui limite pratiquement la validation aux extrémités du processus : expression des besoins pour la spécification puis degré de satisfaction pour l'évaluation. D'autre part, seule la qualité de la réponse aux besoins spécifiés fait l'objet du contrôle. Or tout passage de l'abstrait au concret consiste non seulement à donner corps à ce qui n'était au départ que des potentialités, mais aussi à refermer le domaine des possibilités futures. Comme le soulignent Winograd et Flores [1986]: « Toute ouverture de nouvelles possibilités en ferme d'autres, et ceci est particulièrement vrai avec l'introduction de nouvelles technologies.L'attention aux possibilités de celles qui sont éliminées est aussi importante que la création de nouvelles potentialités. ». La satisfaction des besoins exprimés se paye ainsi par l'impossibilité à répondre ultérieurement à ceux qui n'ont pas été énoncés à temps. Ce prix est parfois si exorbitant qu'il justifierait une reformulation des besoins. Encore faut-il pour cela prendre conscience des bombes à retardement que fabriquent les choix négatifs. Or ce sont, en quelque sorte, des vices cachés qu'il est difficile de diagnostiquer lorsque l'on n'a pas participé directement au processus de fabrication. Pour s'en convaincre, il suffit de se remémorer l'ignorance dans laquelle sont longtemps restés les managers du choix consistant à coder les années sur deux octets.
2. Dans la majorité des cas, le processus se déroule de manière linéaire, partant de problèmes supposés bien définis pour aboutir aux solutions. Cette approche, que l'on peut qualifier de positiviste, repose sur l'hypothèse qu'il est possible de spécifier les problèmes indépendamment des solutions mises en oeuvre pour les résoudre. Contrairement à une approche constructiviste, elle réduit la dialectique problèmes-solutions à une

communication unilatérale. Les démarches itératives, consistant à valider chaque étape par rapport à la précédente, permettent des retours en arrière qui enrichissent la dialectique. Pourtant elles sont aussi largement conditionnées par la spécification initiale des besoins.

2.2. Pertinence de la démarche

Tant que l'informatisation ne vise qu'à améliorer l'efficacité de l'existant, celui-ci constitue une référence stable dans laquelle il s'agit d'insérer le futur système. Les solutions informatiques peuvent alors être recherchées dans un contexte bien défini. Cela permet aux membres de l'organisation d'identifier clairement leurs besoins à partir des manques et des dysfonctionnements qu'ils peuvent observer dans leur vécu quotidien. En outre, ces besoins peuvent être aisément traduits sous forme de fonctionnalités attendues des futures applications. Qui plus est, l'observation de l'existant ouvre aux informaticiens la possibilité de replacer ces fonctionnalités dans leur contexte. Pour résumer, l'espace du problème étant parfaitement défini, la marge d'interprétation laissée aux informaticiens quant à l'élaboration de la solution est relativement réduite. En tout état de cause, la clarté du problème posé laisse toute latitude au donneur d'ordre pour exercer son contrôle lors des phases de validation.

Lorsque l'on se place dans la perspective d'un changement organisationnel, le problème change de nature. Les besoins ne peuvent plus être diagnostiqués à partir de l'observation de l'existant, mais doivent être imaginés dans le cadre du projet organisationnel. Ils en deviennent nécessairement plus flous et donc difficiles à traduire en terme de fonctionnalités précises. Les informaticiens le ressentent bien lorsqu'ils reprochent aux utilisateurs de ne pas savoir exprimer leurs besoins. Le malaise est d'autant plus vivement ressenti que la disparition de l'existant en tant que référence empêche de resituer ces besoins dans un contexte précis.

Par ailleurs, autant il est possible de tronçonner un contexte clair en sous espace, autant il devient très dangereux d'appliquer un tel découpage à un projet d'organisation qui, par essence, présente un caractère systémique très marqué. Plus flou et plus fuyant, l'espace du problème dans lequel se situe l'élaboration du système informatique devient aussi beaucoup plus large. Cela exige une compréhension beaucoup plus globale de l'organisation et de ses différentes dimensions. Mal guidée par des problèmes mal définis, faiblement canalisée par un contexte devenu lui-même très flou, la recherche des solutions laisse alors une très large place à l'interprétation.

Plus fondamentalement, dès lors qu'il s'agit de construire, définition des problèmes et recherche de solutions ne doivent plus se suivre mais s'imbriquer. Toute ébauche de solution soulève, ou permet de formuler, de nouvelles questions. C'est bien pourquoi les informaticiens reprochent aux utilisateurs de changer leurs spécifications en cours de développement. Ce n'est pas que les utilisateurs soient devenus subitement versatiles, mais simplement que besoins et solutions émergent conjointement en cours de construction. Vouloir les dissocier artificiellement en phases distinctes conduit à alourdir et à appauvrir la dialectique qui doit s'instaurer entre les deux. Il s'avère improductif d'aller trop en avant dans la définition d'une solution avant de mesurer l'impact qu'elle peut avoir sur la définition des besoins. Le coût généré est alors assez dissuasif pour contrecarrer le dialogue et donc l'évolution du projet. Face à une solution répondant à ses spécifications, un donneur d'ordre aura du mal à refuser sa validation alors même qu'il prend conscience de besoins nouveaux.

Par ailleurs, l'émergence de nouvelles questions pertinentes ne peut se réaliser que dans l'esprit de personnes ayant participé à la fois à l'analyse et à l'élaboration des esquisses de solutions. Les besoins initialement définis ne devraient constituer qu'un point de départ. Si les validations successives du projet ne portent que sur ces besoins, le contrôle exercé sur les choix effectués par les informaticiens est en fait très partiel.

2.3. Conséquence sur le pilotage des représentations

Contrairement à ce qui se passe lorsque l'on se situe dans le contexte d'amélioration de l'efficacité d'une organisation existante, une optique de changement plus profond implique une imbrication permanente des problèmes de spécification, d'analyse et de conception. Maintenir un enchaînement séquentiel conduit à laisser une large place aux interprétations des concepteurs. Les systèmes informatiques, ainsi élaborés, reposent sur les représentations propres aux informaticiens. De par leur impact sur l'organisation en général et sur le sous-système cognitif en particulier, la reconception des systèmes informatiques exerce une influence déterminante sur les acteurs. A ce titre, elle constitue un puissant levier de pilotage de leurs représentations. Sous cet aspect, le pilotage des représentations échappe alors en grande partie au contrôle des managers pour passer sous celui des informaticiens (figure 8).

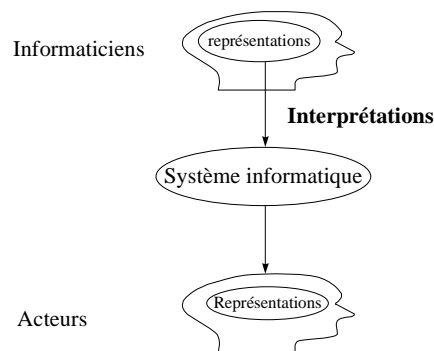


Figure 8 : Pilotage des représentations par les informaticiens

Pour prendre la véritable mesure du danger, il faut réaliser que les informaticiens n'ont généralement aucunement conscience de l'influence qu'exercent ainsi leurs propres interprétations. Leur marge d'interprétation, et donc le risque, se cristallisent principalement sur la séparation des phases de spécifications, d'analyse et de conception du modèle. Réunir les trois revient à adopter une approche constructiviste fondée sur une élaboration simultanée des modèles de l'organisation et de son sous-système informatique. Pourtant, au delà du modèle en tant que résultat, c'est la participation au processus de sa construction qui constitue la source principale d'enrichissement. L'acte de modélisation sert de catalyseur à l'émergence des questions pertinentes que soulèvent le changement organisationnel. Seule une participation conjointe des managers et des informaticiens à la co-construction des modèles permet de prendre en compte l'ensemble des dimensions du changement organisationnel.

Une telle perspective se heurte évidemment à de nombreux obstacles, parmi lesquels figurent les enjeux de pouvoir, les différences culturelles, la recherche d'un retour rapide sur investissement ou simplement les habitudes enracinées. Elle suppose aussi, en préalable, l'adoption d'un langage commun de modélisation.

3. Vers un langage commun de modélisation

La recherche d'un langage capable à la fois de refléter la réalité organisationnelle que l'on veut bâtir et d'être transposé en développements informatiques peut sembler relever de l'utopie tant les différences culturelles et les préoccupations des sphères managériales et informatiques apparaissent différentes. Pourtant, nous tenterons de montrer que les évolutions de ces deux dernières décennies permettent aujourd'hui de remplir les conditions d'une possible convergence. Nous étudierons ensuite plus précisément l'opportunité et le risque que représentent l'émergence d'un standard dans le monde informatique.

3.1. Approches de modélisation : des évolutions convergentes.

3.1.1. La remise en cause des approches fonctionnelles.

Jusqu'au début des années 80, les approches fonctionnelles sont restées largement dominantes aussi bien dans le domaine de la modélisation d'entreprise que dans celui de l'informatique. Dans ce type d'approche, on recherche en premier lieu les fonctions que devra remplir le système ce qui revient à se poser d'abord les questions « pour quoi ? ». Adoptant une démarche analytique classique, ces fonctions sont ensuite découpées en fonctionnalités plus élémentaires. Les entités du futur système sont ensuite définies par un couplage strict avec ce découpage fonctionnel. La structure du système, correspondant aux questions « comment ? », est donc intimement dépendante des réponses apportées aux questions « pour quoi ? ». Les structures fonctionnelles dans les organisations reposent sur ce type d'approches. C'est aussi le fondement de la plupart des systèmes et applications informatiques (figure 9).

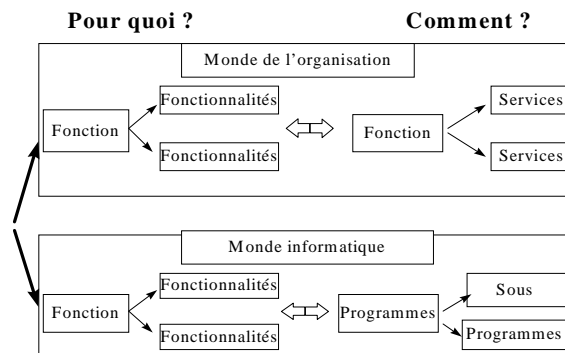


Figure 9 : Les approches fonctionnelles

Intuitivement la démarche semble marquée du sceau du bon sens. Il paraît effectivement logique de partir des questions « pourquoi et pour quoi ? » avant de formuler les questions « quoi et comment ? ». Pourtant, le couplage entre fonctions et structure rend la robustesse de cette dernière complètement dépendante de la stabilité des fonctionnalités. Tout changement substantiel dans la définition des fonctionnalités affecte la structure même du système. Ainsi reproche-t-on, aujourd'hui, aux organisations fonctionnelles de faire preuve d'une grande rigidité face aux évolutions de l'environnement. Dans le domaine informatique, ce type d'architecture explique l'explosion des coûts de maintenance des applications qui peuvent couramment représenter 70% du coût total. Près de 60% de ce montant provenant de modifications de demandes d'utilisateur ou de formats de données [Meyer, 1991].

La recherche de robustesse au changement des modèles et d'adaptabilité des systèmes ont, de ce fait, conduit au développement d'approches rompant avec le couplage strict entre les entités du système et ses fonctions. Nous les intitulerons approches « modulaires »

3.1.2. L'émergence des approches modulaires.

Aucune approche ne peut faire l'économie ni des questions « Pour quoi ? » ni des questions « Comment ? » ni de celles concernant leur articulation. Mais pour gagner en robustesse face au changement, les entités de base du systèmes, que nous nommerons composants, ne sont pas construites pour satisfaire des fonctionnalités précises mais pour remplir des services d'un ordre plus général. Les fonctions du systèmes sont assurées de manière coopérative par des combinaisons de composants. Le couplage entre fonctions et structure est donc réalisé grâce aux liens dynamiques établis entre les entités. Les changements de fonctionnalités affectent alors ces liens sans toucher à la structure. Dans un contexte mouvant, les composants du système peuvent rester relativement stables alors même que les fonctions qui président à leurs combinaisons sont éminemment variables.

C'est bien parce que les activités réalisées dans l'organisation présentent un certain caractère de stabilité qu'elles se sont progressivement imposées comme briques de base dans le domaine de la modélisation d'entreprise. L'aspect dynamique du système peut alors être abordé à travers la participation des activités à des processus construits d'après des critères de finalités. L'approche de la modélisation par les activités et processus aboutit à une conception modulaire permettant de modéliser à la fois les aspects statiques fondant la substance du système (les composants de base) et les aspects dynamiques permettant sa finalisation et son adaptation (les combinaisons entre composants). L'approche objet, dans le monde informatique, procédant des mêmes causes, débouche sur une conception comparable de la modélisation. Les objets représentent d'abord des abstractions de ce qu'est le monde réel, et permettent ensuite de façonner des composants informatiques de base. Comme les activités, ils permettent de définir la partie statique du système. Les fonctions sont alors remplies par collaborations entre objets. Comme celle des processus, la conception de ces collaborations constitue l'aspect dynamique du système. Il doit alors devenir possible de modifier, d'ajouter ou de supprimer des fonctions sans remettre en cause la structure du système.

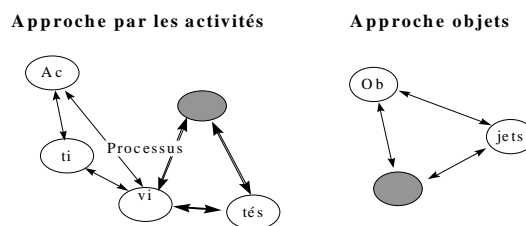


Figure 10 : Les approches modulaires

L'analogie entre les deux approches pourrait être poussée beaucoup plus loin. Par exemple, de même que l'intérieur de l'activité relève du domaine privé de l'acteur, l'état interne de l'objet peut rester caché aux autres objets. Qui plus est, comportement de l'objet et de l'activité doivent être, au contraire, parfaitement visibles (figure 11). L'activité peut donc être considérée comme un objet.

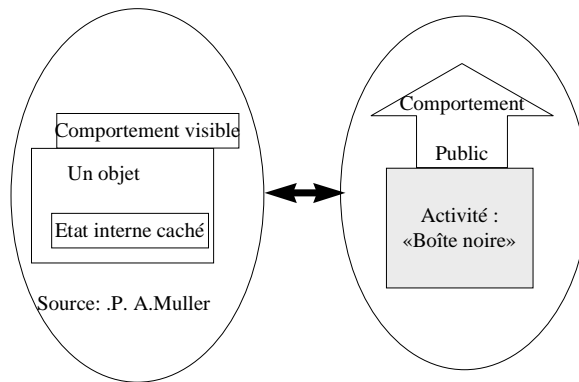


Figure 11. Analogies entre les concepts d'activités et d'objets

En mariant les concepts d'objets, d'activités et de processus au sein de son ontologie d'entreprise TOVE, l'université de TORONTO a déjà prouvé la faisabilité et la fécondité d'une démarche combinant les deux approches.

3.2. L'émergence d'un standard

Si, dans un premier temps, l'approche objet n'a généré que des langages de programmation (Smalltalk, C++, Java...), le besoin de faire reposer leur utilisation sur une démarche spécifique de modélisation s'est rapidement fait ressentir. Il en est ressorti une profusion de méthodes développées dans les années 90 (OMT, BOOCH...). Le regroupement en consortium des principaux acteurs du monde informatique a mis un terme à ces initiatives erratiques en proposant un langage commun de modélisation (Unified Modelling Language). Issu du monde informatique, UML présente néanmoins des caractéristiques intéressantes pour une sphère managériale en quête d'outil de formalisation.

- En tant que langage de modélisation, UML se veut largement détaché des contraintes de développement informatique. Ce détachement, obtenu grâce à un haut niveau d'abstraction, était d'ailleurs une condition nécessaire pour réunir des acteurs aux intérêts divergeants. Il en résulte un langage suffisamment généraliste pour être transposable en dehors de la sphère qui l'a vu naître. C'est d'ailleurs une voie en cours d'exploration au sein même de la communauté UML comme l'atteste la tentative de développement d'une branche spécifique à la modélisation du fonctionnement des organisations (UML for Business Process).
- Compte tenu des enjeux financiers sous-jacents, le consensus sur UML n'a pu être obtenu qu'au prix de l'intégration d'éléments divers provenant des différents membres du consortium. Ce processus, plus additif que soustractif, entraîne une certaine hétérogénéité de l'ensemble, mais il permet aussi de couvrir une très large palette de besoins et d'utilisations. Cela implique que la standardisation n'est pour l'instant que très relative et qu'il est probable que son renforcement vienne ultérieurement de l'usage.
- Le poids des protagonistes qui lui ont donné naissance (de Microsoft à IBM) permet à UML de s'imposer d'emblée comme un standard de fait dans le monde informatique. Si le besoin se faisait ressentir de manière pressante et si les alternatives manquaient, il n'est pas inenvisageable qu'UML s'impose bien au delà de sa sphère d'origine.

L'évolution organisationnelle dans le sens d'une autonomie accrue des acteurs déplace le problème de la coordination du contrôle des actions vers le pilotage des représentations des acteurs. En structurant à la fois le partage et les échanges de données, le système informatique joue un rôle essentiel dans ce pilotage. Les managers ne peuvent espérer piloter ces représentations qu'à la condition de garder la maîtrise du système informatique, et donc de participer activement à sa conception. Cette participation active, au côté des informaticiens, nécessite le partage d'un langage commun de modélisation.

Le contexte de changement croissant dans lequel évoluent les organisations a entraîné, dans les deux sphères, une évolution comparable en matière d'approche de modélisation. Dans les années 80, l'approche par les activités et processus et celle des objets marquent une rupture avec la modélisation fondée sur les fonctionnalités du système à modéliser. Toutes deux reposent d'abord sur la définition des composants stables des systèmes puis sur la réalisation des leurs fonctions à partir de l'établissement de liens dynamiques entre ces composants. Au delà de cet aspect fondamental, les deux approches présentent de nombreuses autres analogies. Nous avons mis en lumière la similitude entre le caractère privé de l'intérieur de l'activité et son pendant dans la démarche objet, dénommé encapsulation, (Figure 11). On peut également citer le polymorphisme dans l'approche objet qui correspond au fait qu'une même opération peut être réalisée de manières différentes en fonction des caractéristiques et du contexte propre à chacun des objets. On retrouve ce polymorphisme au niveau des activités puisqu'elles remplissent leurs missions en fonction de leur contexte local. On aurait pu aussi pousser les comparaisons en ce qui concerne les concepts « objet » de hiérarchisation, d'héritage et d'agrégation des composants des systèmes susceptibles aussi d'organiser les activités au sein de processus et de sous-processus.

Néanmoins, tant que les approches restent focalisées sur des finalités aussi spécifiques que l'élaboration de systèmes de calcul de coûts ou la réalisation de programmes informatiques, ces analogies ne suffisent pas à bâtir un langage commun. Par contre, dès l'instant où l'on s'éloigne d'objectifs en terme d'outils pour s'orienter vers la recherche d'un langage de modélisation détaché de finalités trop précises, la montée en abstraction qui accompagne ce mouvement permet de transformer de simples analogies en véritables convergences. Or, c'est bien à une telle évolution que l'on assiste actuellement. L'approche par activités déborde aujourd'hui largement de ses problématiques initiales, tournées vers l'élaboration d'outils de gestion comme l'ABC, pour tenter de modéliser plus globalement l'organisation et son fonctionnement. De même, à travers UML, l'approche objet tente de se libérer des objectifs et contraintes techniques de programmation. Se plaçant au niveau des racines et non plus à celui des fruits, le rapprochement devient alors envisageable.

Dès lors, si la perspective d'un langage commun de modélisation peut-être considérée comme une utopie, c'est dans le sens que donne Varela [1989] à ce mot : « non pas l'irréalisable mais l'irréalisé ». La question est alors de savoir si les « managers » seront partie prenante de la réalisation où s'ils en laisseront le monopole aux informaticiens.

Références bibliographiques

- FLORES F. et WINOGRAD T. (1986) *L'intelligence artificielle en question* Ablex Publishing Corporation édition française PUF (1989)
- JACOBSON J (1993) *Le génie logiciel orienté objet* édition française, Addison-Wesley
- JACOT J.H. et MICAELLI J.P.(1996) : « La rationalisation cognitive au cœur des nouvelles formes d'organisation » dans *Cohérence, Pertinence et Evaluation* groupe ECOSIP, Economica Paris

- LEMOIGNE J.L.(1996) : « Les deux sources de la performance des organisations : incohérence du contrôle, impertinence de l'intelligence » *Cohérence, Pertinence et Evaluation* groupe ECOSIP, Economica Paris
- LORINO P.(1996) : « Le pilotage de l'entreprise : de la mesure à l'interprétation » dans *Cohérence, Pertinence et Evaluation* groupe ECOSIP, Economica Paris
- LORINO P. (1995) : *Comptes et récits de la performance* Les éditions de l'organisation
- MEVELLEC P (1996) « Modèle d'entreprise et système de calcul de coûts » dans *Cohérence, Pertinence et Evaluation* groupe ECOSIP, Economica Paris
- MEYER B. (1991) *Conception et programmation par les objets* InterEditions Paris
- MINTZBERG (1982) *Structure et dynamique des organisations* édition française Editions d'Organisation, Paris.
- MORIN E. (1977) *La méthode I. La nature de la nature* Points Seuil, Paris
- MULLER P.A. (1997) *Modélisation objet avec UML* Eyrolles, Paris
- MINSKY M. (1988) *La société de l'esprit* édition française InterEditions, Paris
- PIAGET J. (1970) *L'épistémologie génétique* Que sais-je . PUF, Paris
- PRINCE V. (1996) *Vers une informatique cognitive dans les organisations* Masson
- REIX R. (1997) *Systèmes d'information et management des organisations* éditions Vuibert
- RUMBAUCH J. (1996) *O.M.T. Modélisation et conception orientées objet* Prentice Hall édition française Masson
- TABATONI P. et JARNIOU P. (1975) *Les systèmes de gestion* Presses Universitaires de France, Paris
- VARELA F. J. (1989) *Connaître les sciences cognitives* édition française Seuil