

DE L'INFORMATIQUE
Savoir vivre avec l'automate

Michel Volle

17 avril 2006

Introduction

« The question “What can be automated?” is one of the most inspiring philosophical and practical questions of contemporary civilization » (George Forsythe, « Computer science and education », in *Information processing 68*, North-Holland 1969, [60] p. 92)

« Computer science answers the question “What can be automated?” » (Donald E. Knuth, « Computer Programming and Computer science », in *The Academic Press Dictionary of Science and Technology*, 1992, [103])

L’informatisation est le phénomène le plus important de notre époque. Certes, ce n’est pas le plus spectaculaire : il n’explique ni l’explosion démographique de l’espèce humaine, ni le changement climatique ; nos paysages, nos maisons, nos villes, nos voitures, nos équipements ménagers, ont été conçus sous les systèmes techniques antérieurs (Gille [66]). Cependant ce qui importe n’est pas ce qui se voit, mais plutôt ce qui s’enracine, se déploie et oriente le reste.

Le système technique informatisé est né dans les dernières décennies du XIX^e siècle (voir page 68). Il a, dans les années 1970, pris le dessus sur le système technique mécanisé. La mécanique et la chimie n’ont pas disparu bien sûr, mais la synergie fondamentale est désormais celle qui associe la microélectronique au logiciel.

En prenant en charge dans les usines puis dans les bureaux la part répétitive du travail, l’*automate* a rompu le lien qui, dans l’économie mécanisée, reliait l’emploi à la production et avait, malgré les crises et les guerres, procuré un équilibre endogène. Le réseau lui a conféré l’ubiquité. Il a transformé la relation entre le coût de production et la quantité produite : l’essentiel du coût d’un produit réside dans sa conception et sa mise en place, le coût marginal est devenu pratiquement négligeable (c’est évident dans le cas des logiciels et de la microélectronique et cela s’étend aux produits dont la production est fortement automatisée). La fonction de production des entreprises en a été transformée ainsi que les conditions de la concurrence ; la mondialisation de l’économie en résulte (Volle [213]). Par ailleurs le système d’information, s’appuyant sur l’automate, est devenu le *langage* de l’entreprise.

Cette évolution déconcerte des institutions qui avaient été construites

pour l'économie antérieure : les nations dont la prospérité s'est fondée sur la mécanique et la chimie connaissent toutes, aujourd'hui, une crise de l'éducation, de l'emploi, de la santé, de la retraite. Il ne suffit plus pour l'économiste de raisonner en termes d'équilibre entre offre et demande : il doit remonter, vers l'amont, jusqu'à la fonction de production et aux besoins.

* *

Un tel changement suscite naturellement la plus grande confusion dans les idées, dans les valeurs. Je revois ce directeur général qui, alors que son entreprise recevait déjà une part significative des commandes *via* le site Web, disait en se pavanant comme un dindon « Moi, je ne crois pas à l'Internet ». Je revois cet économiste, porteur d'un nom célèbre, dire avec un mépris accablant « la fonction de production, c'est un concept d'ingénieur ; ce qui compte, c'est l'offre et la demande¹ ».

Nos entreprises, nos institutions s'informatisent à l'aveuglette. Elles sont comme une personne qui avancerait à reculons, poussée par une main posée sur sa poitrine, et trébucherait sur le moindre obstacle. Lorsqu'on les examine, et une fois surmontée l'impressionnante technicité du vocabulaire et des méthodes, on découvre des absurdités qui surprennent ainsi que les résistances, plus surprenantes encore, qui s'opposent à ce qu'on les rectifie.

Telle entreprise s'est organisée de façon à ne pas pouvoir connaître ses clients ; telle autre, faisant confiance à l'héritage de son histoire, a depuis longtemps renoncé à réfléchir à son processus de production ; telle autre encore laisse son langage, ses classifications, se détériorer au hasard des applications informatiques et des dialectes locaux. Certaines, dans les services, ne sauraient définir ni leur produit, ni moins encore les critères selon lesquels on peut évaluer sa qualité. Alors que la moitié du temps des salariés se passe devant le couple écran-clavier (voir page 201), la plupart des entreprises ne se soucient guère de la relation entre l'utilisateur et son poste de travail.

* *

Que le passage d'un système technique à l'autre provoque des désordres, qu'il mette en question les institutions antérieures, les rapports sociaux, qu'il pose des questions de savoir-faire, de savoir-vivre et suscite le désarroi, cela n'a rien de nouveau. Il en fut de même à la Renaissance et lors des débuts de l'industrialisation. Les guerres de religion, les deux guerres mondiales du XX^e siècle, peuvent s'interpréter comme des épisodes particulièrement violents d'une crise d'adaptation.

Mais pour tempérer cette crise, pour en sortir au plus vite, il faut avoir compris ce que le nouveau système technique a de spécifique. La mécanique et la chimie avaient, en s'aidant d'engrenages et de moteurs, permis de soulager l'effort *physique* que la production réclame à l'être humain. L'automate vient maintenant soulager l'effort *mental* en prenant en charge la surveillance et l'attention prolongées, les vérifications et calculs répétitifs,

1. Il ignorait apparemment que la fonction d'offre, quand elle existe, s'obtient par un calcul à partir de la fonction de production et du coût des facteurs ; dans le cas du monopole naturel, elle n'existe pas.

les transcriptions, les classements et recherches documentaires etc. Dans le processus de production il s'enlace avec le travail humain selon une synergie intime et délicate : la frontière qui les articule est une interface d'une extrême complexité.

Du coup ce ne sont pas seulement les institutions qui sont en question, ni la fonction de production des entreprises : c'est la façon dont nous agissons, dont nous pensons, dont nous nous organisons ; c'est la relation entre notre pensée et notre action ; c'est notre rapport à la nature ; c'est l'ordre que nous mettons dans les valeurs que nous entendons promouvoir.

* *

On considère l'*informatique* comme une activité *technique*. C'est ce qu'elle est en effet, puisqu'elle concerne l'action et demande un savoir-faire. Mais elle constitue aussi une innovation *intellectuelle* dont l'ampleur, la profondeur et les implications se comparent à celles de l'invention de l'alphabet par les Phéniciens vers 1000 avant notre ère (Bonfante et coll. [18]), puis des mathématiques et de la philosophie par les Grecs.

S'écartant de la question « qu'est-ce que c'est ? », qui faisait de la *dé-finition* le premier pas de la connaissance, elle pose en effet la question « comment faire ? » et met ainsi l'Action sur le trône de la pensée, où elle supplante l'Être qui l'occupait depuis Parménide (VI^e et V^e siècles avant JC).

Donner à la pensée une finalité essentiellement *pratique*, c'est un changement de perspective aussi radical que celui qu'introduisit Galilée (1564-1642) lorsqu'il substitua, dans la construction de la connaissance, la démarche expérimentale à l'argument d'autorité « *magister dixit* ». Ce changement est confronté à des oppositions aussi obstinées que celles que la démarche expérimentale a rencontrées.

Les absurdités que l'on découvre dans les entreprises, dans les institutions, s'expliquent en partie par l'inertie des structures et la sociologie des corporations mais surtout par des conceptions du monde et de la réalité, par des formes de pensée, des échelles de valeur héritées de l'histoire et d'autant moins maîtrisées qu'elles sont plus prestigieuses. Lorsqu'on touche aux structures en place, aux intérêts des corporations, cela suscite certes des réactions violentes ; mais celles que l'on provoque lorsque l'on travaille sur les distinctions entre le réel et l'imaginaire, entre l'essentiel et le secondaire, sont plus violentes encore car elles touchent à l'échafaudage souvent bancal des valeurs sur lesquelles chacun appuie son propre discernement, ses propres priorités. Les obstacles de nature philosophique, métaphysique, sont plus difficiles à surmonter que les obstacles institutionnels : nous en fournirons plusieurs exemples au cours de cet ouvrage.

* *

Pour examiner la relation entre l'action, la pensée et les valeurs chacun dispose du laboratoire que constitue sa propre personne où se déroulent des phénomènes d'une riche complexité. Mais comme ils sont quotidiens, familiers, intimes, on ne croit pas qu'ils méritent l'attention du chercheur. Il est pourtant utile, avant de s'interroger sur ce qui se passe dans l'entreprise,

de percevoir en nous-mêmes, à l'échelle de l'individu, comment s'enchaînent la perception, l'expérience, le discernement, le raisonnement, la décision et l'action (voir page 210).

Entre l'action individuelle et l'action organisée existe cependant le même écart qu'entre le bricolage - utile, voire indispensable - que chacun peut faire à la maison et la production industrielle. Le lieu géométrique de l'action organisée, le lieu où les êtres humains transforment la nature pour se procurer le bien-être, c'est en fait *l'entreprise*²; l'informatique étant tout entière orientée vers l'action, c'est dans l'entreprise qu'elle aura son plein effet.

Envisager l'entreprise sous cet angle conduit à effacer des frontières que l'histoire a gravées dans le droit : si l'on définit l'entreprise par la production de bien-être pour les consommateurs ou, pour parler comme les économistes, d'*utilité*, ce concept inclura en effet les services publics et une part de ce que l'on nomme administration (voir page 156). Cela conduit aussi à la libérer du carcan financier comme de la grille sociologique dans lesquels trop de raisonnements l'enferment.

* *

Je ne prétends pas publier ici un traité complet sur l'informatique : ce sujet inépuisable est bien de ceux à propos desquels on peut dire $\acute{o} \beta\acute{\iota}\omicron\varsigma \beta\rho\alpha\chi\acute{\upsilon}\varsigma, \acute{\eta} \delta\grave{\epsilon} \tau\acute{\epsilon}\chi\eta \mu\alpha\kappa\rho\acute{\eta}$, l'art est long mais la vie est courte. Cette recherche ne s'arrêtera pas avec ce livre et quand je mourrai elle ne sera pas achevée.

Je l'ai intitulé *De l'Informatique* pour indiquer le plus simplement possible de quoi il est question³. On verra d'ailleurs (page 198) que l'on peut utilement donner au mot « informatique » un sens plus proche de son étymologie que de l'usage courant. Le sous-titre « savoir vivre avec l'automate » précise de quoi il est question : on peut lire aussi « savoir-vivre », avec un trait d'union.

Je l'ai construit comme une mosaïque de développements simples, le sens devant résulter de leur agencement. Cette technique m'a été suggérée par la juxtaposition de tissus aux dessins divers dans les kimonos japonais anciens et par les *Mémoires* de Saint-Simon : elle facilite la mise en rapport de faits méconnus, de points de vue peu fréquentés, de notions étrangères à la mode ou que l'on croit banales mais qui méritent pourtant l'attention.

Quel que soit le talent individuel chacun ne peut avoir sur l'informatique qu'une vue partielle. Ainsi Donald Knuth lui-même s'est focalisé sur *l'art de la programmation*, ce qu'indique d'ailleurs exactement le titre de son ouvrage monumental (Knuth [105]). Dans les travaux des historiens, et fussent-ils de qualité, on constate des lacunes : tel accorde plus d'importance au matériel

2. Il existe certes des entités prédatrices qui, tout en usurpant le nom d'« entreprise », se sont fixé de tout autres buts que l'accroissement du bien-être des êtres humains (Verschave [208]). Qu'une mission soit trahie ne change cependant rien à son énoncé : voir page 167.

3. Un titre court est signe de modestie : un manuel scolaire s'intitule *Mathématiques* sans que son auteur prétende avoir épuisé le sujet. Certains estiment pourtant que la modestie exigerait un titre contourné commençant par quelque chose comme *Prolégomènes à une introduction à...*

qu'au logiciel (Ceruzzi [35]); tel autre néglige les logiciels d'usage courant sur le micro-ordinateur (Campbell-Kelly [29]); presque tous, s'arrêtant au seuil de l'époque actuelle, n'effleurent pas certains sujets qui sont pour nous brûlants.

Mais comme il est impossible de décrire un objet réel, concret, sans le considérer à partir d'un point de vue particulier, il serait vain d'en tenir rigueur à un auteur : lorsqu'on s'enquiert, auprès des personnes qui font ce genre de reproche, de la description qu'elles jugeraient « objective » elles révèlent d'ailleurs souvent, et de façon réjouissante, un point de vue encore plus étroit que celui qu'elles critiquent.

Mon propre point de vue est celui des *utilisateurs* de l'informatique, de sa *maîtrise d'ouvrage*. Il est certes partiel mais ni plus ni moins légitime que celui des théoriciens de l'informatique, des experts en langages de programmation, des architectes, des programmeurs.

Il se peut même que dans la conjoncture actuelle il soit l'un des plus *pertinents*. L'informatique est pour les entreprises une ressource puissante dont la maîtrise est digne de focaliser l'attention du *stratège*, de son dirigeant suprême. Cependant la réflexion reste en retard car beaucoup de nos dirigeants sont comme ces généraux français qui, dans les années 1930, préféraient les chevaux aux blindés. Il en résulte des systèmes d'information mal conçus, une informatique mal utilisée - d'où un malaise et, parfois, des catastrophes qui scandalisent.

La responsabilité d'un sinistre incombe pratiquement toujours à la maîtrise d'ouvrage, à l'entité utilisatrice : elle n'aura pas su dire ce qu'elle voulait ni définir ses priorités, elle aura modifié en cours de route l'expression de ses besoins, elle aura voulu régler un problème « politique » en se cachant derrière un alibi technique, elle aura manqué de courage devant un fournisseur défaillant, elle n'aura pas voulu écouter ceux qui tentaient de l'alerter.

* *

Certains économistes ont mis en doute l'efficacité de l'informatique (voir page 181), mais leurs analyses sont entachées par un défaut de perspective : on ne doit pas reprocher à une technique encore nouvelle les errements que provoque non cette technique elle-même, mais le manque d'expérience de ses utilisateurs. Il ne convient pas non plus de lui reprocher la difficulté de l'apprentissage même s'il faut savoir exiger une bonne ergonomie : apprendre à lire et à écrire est des plus pénibles, pourtant personne ne songe à en tenir rigueur aux inventeurs de l'alphabet.

Si l'informatique était inefficace, il suffirait à une entreprise de la supprimer pour se porter mieux. Or c'est tout le contraire : une banque, une société d'assurance, un transporteur aérien, un opérateur télécom etc. qui renonceraient à l'informatique cesseraient du même coup d'exister. Ce fait, d'une évidence massive, suffit pour révoquer en doute des affirmations fondées sur une utilisation imprudente de la statistique⁴.

4. Lorsqu'on veut évaluer l'efficacité d'une technique nouvelle il faut utiliser l'approche monographique et non la statistique : les totaux et les moyennes additionnent des utilisations

C'est donc à bon droit que la presse informatique et les fournisseurs publient des *success stories*. Mais les *failure stories* sont plus instructives encore car elles indiquent les pièges à éviter. Elles ont d'ailleurs parfois un comique qui fait sourire et réveille l'attention. Dans les églises romanes, les mosaïques qui représentent l'enfer sont plus mouvementées, plus intéressantes que celles d'en face : à en croire ces dernières, on s'ennuie terriblement au paradis...

Ce n'est donc pas par mauvais esprit que j'accorderai une place à l'évocation des erreurs et des malfaçons. Même si elles nous font enrager je crois leur évocation utile, instructive, et comme elle est parfois amusante nous aurions grand tort de nous en priver. Je ne citerai aucun nom propre et ferai en sorte que l'on ne puisse pas identifier les entreprises dont il s'agit.

* *

Je me suis intéressé à l'informatique à proportion des difficultés que j'ai rencontrées en l'utilisant ou en conseillant mes clients. Il m'est pratiquement impossible en effet de me servir d'un outil dont je ne comprends pas le fonctionnement et dont je ne perçois pas comment il a pu être conçu. Ce travers m'a souvent rendu ridicule, par exemple lorsque j'ai appris à conduire une voiture ou à utiliser un traitement de texte ; mais il contraint à une *écologie de la pensée* qui, sur le long terme, porte des fruits : le monde s'éclaire lorsque son exploration progresse⁵.

« Comprendre », cela ne va pas jusqu'à reproduire exactement par la pensée le fonctionnement de l'outil car celui-ci est soumis à d'innombrables contraintes physiques : il s'agit seulement de ramener cet artefact aux principes dont ses concepteurs ont voulu assurer la réalisation pratique. On peut par exemple « comprendre » le fonctionnement du moteur à explosion à partir de quelques éléments de chimie des hydrocarbures et de mécanique, tout en faisant l'impasse sur les questions de métallurgie qu'il a fallu pourtant traiter pour mettre au point des alliages ayant les qualités nécessaires. De même, lorsqu'on examine un logiciel, on fera souvent l'impasse sur les « couches basses » diablement complexes où des électrons sont contraints à fournir la matière première des bits, octets, données, bases de données, fichiers et programmes.

Pour comprendre comment il a pu paraître nécessaire et naturel de concevoir un artefact, il faut revenir à l'histoire et reconstituer la situation qui fut celle de ses inventeurs. Les enseignements qu'apporte cette enquête n'éclairent pas le seul passé : en faisant apparaître la dynamique du rapport entre le cerveau humain et la nature, puis celle (à la fois institutionnelle et culturelle) de l'innovation, elle invite à concevoir le présent comme un point d'une trajectoire dont on pourra anticiper l'évolution, fût-ce de façon qualitative et dans les grandes lignes.

Il est intéressant aussi d'observer ce qui se passe en soi-même lorsqu'on apprend à utiliser un nouvel outil informatique, à le dépanner etc. qu'il

teurs plus ou moins adroits alors qu'il ne faut considérer que les plus adroits, ceux dont l'habileté anticipe l'état de l'art futur.

5. Un exemple éminent de ce type de démarche est celui de Joseph Needham [145].

s'agisse d'un traitement de texte, d'un tableur, d'une imprimante, d'une connexion ADSL, d'un réseau ou d'une grande application de l'entreprise. Pour passer de la complète impuissance à la pratique habituelle il faut s'approprier soi-même⁶ selon un tâtonnement par essais et erreurs lors duquel l'image mentale de l'outil s'ajuste jusqu'à devenir assez exacte pour que l'on puisse utiliser celui-ci sans trop de mal. Ces épisodes d'apprentissage durent quelques heures, jours ou semaines, puis s'effacent de la mémoire comme si l'on était passé d'un saut de l'ignorance à la maîtrise. Il est bon cependant de se les remémorer si l'on veut comprendre comment l'informatique peut être assimilée, comment il convient de se comporter avec un débutant.

* *

Ces enquêtes demandent du travail mais elles sont tellement passionnantes que je m'étonne de ne pas voir davantage de personnes les pratiquer comme *hobby*. L'attitude la plus fréquente, semble-t-il, est au contraire d'oublier nos apprentissages pour faire comme si les outils dont nous disposons étaient naturels : il est si simple d'utiliser un moteur que s'interroger sur son fonctionnement et son origine paraît superflu. Ceux qui ont cette attitude savent sans doute qu'il y a eu des innovations mais ils ne le « réalisent » pas, *ils font comme si l'innovation n'était pas réelle*, ils vivent comme si ces outils avaient toujours existé dans l'état où nous les connaissons et devaient persister dans ce même état.

On rencontre souvent de ces personnes qui nient l'innovation, la nouveauté : « Au fond, disent-elles, rien n'a changé et le discours sur les nouvelles technologies a été très excessif ». Il faut les secouer pour les sortir de ce songe : le disque compact n'a-t-il pas supplanté le microsillon en vinyle ? le téléphone mobile, si rare dans les années 1980, n'est-il pas aujourd'hui omniprésent ? le PC portable et la baisse du prix des ordinateurs n'ont-ils pas changé notre rapport à l'informatique ? l'Internet ne procure-t-il pas une *ubiquité logique* qui, voici vingt ans, relevait de la science fiction ? et que dire, si l'on remonte dans l'histoire, du moteur à explosion, du moteur électrique ?

Il faut leur accorder que la « nouveauté » est relative. On s'inquiète des prothèses que l'informatique propose : mais depuis des siècles nous portons des lunettes, qui ne sont pas une prothèse négligeable. On s'inquiète de l'abondance des textes publiés sur l'Internet : mais depuis des siècles nous sommes submergés par la masse des textes publiés, et nous sommes loin de pouvoir tout lire.

Ceux qui nient l'innovation, fait pourtant indéniable, ne veulent-ils pas plutôt dénoncer le discours qui prétend que la technique pourrait « changer l'homme » en mieux ? Peut-être aussi veulent-ils dire : « Non, l'être humain n'a pas changé, à preuve : je suis toujours aussi malheureux ». Mais ce sont là de tout autres affaires ! Si le bien-être que procurent des outils efficaces n'est pas indifférent, il ne suffit pas pour donner un sens à la vie : c'est ailleurs qu'il faut chercher la source de la sagesse et du bonheur.

6. Voir par exemple « S'approprier à un nouveau logiciel », www.volle.com/travaux/latex.htm.

Derrière l'indifférence à l'innovation comme derrière son exaltation se devine l'adhérence à des valeurs implicites. Certains esprits sont ainsi protégés contre toute surprise même s'ils prétendent, conformément à la mode, pratiquer la « remise en question permanente ».

Si l'on cherche à comprendre comment « cela » fonctionne, comment des êtres humains ont pu concevoir « cela », on sera au contraire souvent surpris et contraint de pénétrer des univers mentaux dont on ignorait tout. On découvre alors, derrière des mots comme « entreprise » ou « informatique » que la banalité a comme recouverts de poussière, des phénomènes dont le discours habituel ne rend pas compte parce qu'ils sont jugés soit inexistantes, ce qui est faux puisqu'il s'agit de faits vérifiables, soit négligeables, ce qui peut et doit se discuter.

Ces phénomènes pivotent tous autour de l'articulation entre l'automate et l'être humain ou, pour utiliser dès maintenant un vocabulaire que nous introduirons, entre l'« automate programmable doué d'ubiquité » (APU) et l'« être humain organisé » (EHO) (voir page 200). Or il est dans la nature des spécialités de se détourner d'une telle articulation, toujours quelque peu obscure, pour dévaler la pente vers la clarté de leurs concepts familiers. Celui que cette articulation intéresse sera contraint, par contre, à considérer chaque spécialité comme une boîte à outils où il prendra à l'occasion les instruments utiles à sa démarche. Il devra se faire tour à tour économiste, historien, physicien, linguiste, sociologue, philosophe - et, bien sûr, informaticien. C'est là s'exposer à un rejet par chacune des corporations ainsi effleurées : si le mot *interdisciplinarité* est à la mode, celui qui la met en pratique ne sera jamais le bienvenu !

* *

Considérer l'entreprise sous l'angle de son *action*, c'est un point de vue parmi d'autres mais c'est peut-être le plus salubre car il est bon, lorsqu'on aborde une institution, de s'enquérir en tout premier de *ce qu'elle fait* ou encore de ce qu'elle *produit*.

Parmi les institutions l'entreprise est précisément celle qui a pour spécialité la *production*, sa mission étant de rendre la nature hospitalière pour l'être humain. Le monde de l'action, dans lequel elle baigne entièrement, inclut la pensée qui est une action différée tout comme l'investissement est une production différée.

Mais on ne peut pas séparer l'action des *valeurs* qu'elle incarne dans le monde. En première analyse, l'entreprise ne sert qu'une seule valeur, l'efficacité : il s'agit de faire au mieux, avec les ressources dont elle dispose, pour procurer au consommateur le bien-être matériel. Il serait stupide de renoncer à l'efficacité, de gaspiller des ressources ; mais ni l'efficacité, valeur purement technique, ni le bien-être matériel qui en est le fruit ne suffisent à satisfaire toutes nos aspirations : comme le disait Keynes, l'économie est nécessaire mais non suffisante (Dostaler [49]).

Cependant pour pouvoir produire l'entreprise entre nécessairement en relation avec les êtres humains que sont ses clients et ses salariés. Alors elle rencontre d'autres valeurs que celle de l'efficacité : l'équité des lois et des

règles (Rawls [170]), la qualité de la relation humaine enfin dont l'éventail s'étale du mépris au respect, de la négation à l'affirmation de l'humanité commune.

Même si l'efficacité est pour l'entreprise la valeur première, elle ne peut donc pas s'y limiter : nous verrons d'ailleurs que le respect envers l'être humain est une des conditions de l'efficacité elle-même (voir page 177).

* *

Le langage sert bien sûr à communiquer mais, tout comme l'architecture qui peut aussi bien ériger une forteresse que construire un pont, il sert aussi à se défendre et à attaquer. Ici l'obstacle ne réside pas dans les revendications d'un moi confus et souffrant, mais dans celles des corporations⁷ : les dirigeants, milieu social dont le théâtre est extérieur à l'entreprise ; les *managers*, qui se partagent la légitimité et surveillent jalousement ses frontières ; les agents opérationnels, divisés en métiers et spécialités etc. Chaque spécialité utilise un vocabulaire spécifique : c'est sans doute nécessaire à l'action professionnelle, qui exige la précision, mais cela sert aussi de signe de reconnaissance, de mot de passe, d'arme pour tenir l'autre à distance.

Les corporations se jalourent et souvent se méprisent. Des mots par eux-mêmes neutres comme « technicien », « intellectuel », « informaticien », « ingénieur » etc. reçoivent à l'occasion une connotation tellement péjorative que l'on doit, pour s'en libérer, préciser à chaque fois en quel sens on les utilise.

Alors qu'il faudrait une coopération, un dialogue entre diverses spécialités, l'entreprise est souvent l'enjeu d'un conflit entre des réseaux qui la parasitent. On se plaît à caractériser notre époque comme celle de l'« économie de l'information » ; mais l'entreprise y rencontre, au delà des exigences de la technique et du marketing, celles d'un « commerce de la considération », d'une « économie du respect » (voir page 177) sans laquelle l'économie de l'information ne pourra pas tenir ses promesses ni le système d'information contribuer à l'efficacité.

* *

Parmi les questions de savoir-vivre certaines sont triviales, comme la façon dont on doit utiliser le téléphone mobile dans un lieu public ; d'autres sont moins aisées à discerner. Il en est ainsi de la façon dont nous devons aujourd'hui penser l'informatisation et en parler, dont nous devons agir envers elle. Cette question se pose à chacun dans des termes correspondant à son rôle : à l'utilisateur « de base », au *manager*, au dirigeant, à l'informaticien.

Un des premiers obstacles réside dans le vocabulaire. Un mauvais sort - tellement systématique que l'on pourrait croire qu'il a été délibérément organisé - a rempli le langage de l'informatique de « faux amis », de termes qui, dans la langue courante, sont entourés de connotations étrangères à leur acception technique précise (voir page 28). Il en résulte, entre les in-

7. Nous utilisons ce mot, par commodité, pour désigner le petit monde que forme une spécialité professionnelle.

formaticiens et le reste du monde, une difficulté de communication qui m'a souvent paru presque insurmontable.

Prenons par exemple le mot « abstraction ». Pour l'informaticien, l'abstraction est ce qui permet de ne pas avoir à programmer les opérations physiques que réalisent automatiquement les couches basses de la machine (gestion de la mémoire, reprise en cas d'incident etc.). Pour l'utilisateur, l'abstraction consiste à sélectionner les attributs qui, dans une base de données, représenteront un être du monde de la nature.

Ainsi vous prononcez une phrase qui vous paraît parfaitement claire mais le regard stupéfait de l'informaticien signale qu'elle évoque pour lui quelque chose d'incompréhensible, de très compliqué, éventuellement d'absurde. Parle-t-il ? C'est vous qui ne comprenez rien, noyé entre des mots courants qu'il utilise dans un autre sens que le vôtre et des acronymes et termes techniques qui désignent des choses dont vous ignorez tout.

On peut apprendre la langue des informaticiens : cela permet de remplir l'office d'interprète. On comprend alors pourquoi les relations entre l'informaticien et l'utilisateur sont si souvent du type « chien et chat » : l'hostilité entre les deux espèces s'explique par le fait qu'un même signal (remuer la queue, faire le dos rond, ronronner) exprime chez l'un le contentement, chez l'autre l'agressivité.

Le président-directeur général, le directeur général, les DGA, ne comprennent pas grand chose lorsqu'un informaticien intervient devant le comité de direction. Mais en un tel lieu il serait malvenu de s'avouer ignorant et de demander une explication. Ils opinent donc, et décident, avec l'assurance que leur procure le sentiment de leur légitimité : mais si la décision est judicieuse, ce sera par hasard.

* *

Nota Bene 1 : Le vocabulaire de l'informatique abonde en barbarismes comme « orienté objets », « modèle métier » etc. S'opposer ici à l'usage serait nager à contre-courant. J'utiliserai donc ces expressions bien que je ne les aime guère.

Nota Bene 2 : À ce livre est associé un lexique. Pour éviter d'alourdir un livre qui comportait déjà beaucoup de pages je l'ai mis sur la Toile, à l'adresse www.volle.com/ouvrages/informatique/lexique.htm.

Remerciements

Cet ouvrage s'appuie sur un cours donné à l'Université Libre de Bruxelles, à l'aimable invitation d'Isabelle Boydens, pendant les années 2002-2003 et 2003-2004.

Je dois remercier les institutions et les personnes qui m'ont aidé à explorer l'informatique. Nous n'étions pas toujours d'accord et certaines de ces personnes ne partagent certainement pas l'analyse présentée dans cet ouvrage, mais cela n'enlève rien à ma dette envers elles.

Je remercie d'abord l'INSEE, où j'ai fait mes premières armes de statisticien, de chercheur, de programmeur et d'utilisateur de l'informatique, et où j'ai acquis ma formation professionnelle de base.

Je remercie France Telecom, qui en m'accueillant au CNET, son centre de recherche, m'a permis de renouer avec mon premier amour, la physique, et d'explorer l'univers des TIC en construisant des modèles économiques.

Je remercie les ingénieurs d'Arcome et d'Eutelis, entreprises que j'ai créées dans les années 1990. Je remercie les clients de ces entreprises et en particulier Air France, l'ANPE et Bouygues Telecom, avec lesquelles j'ai eu l'occasion de participer à l'organisation de la maîtrise d'ouvrage.

Je remercie les membres du Club des maîtres d'ouvrage des systèmes d'information (www.clubmoa.asso.fr), qui m'ont permis d'enrichir et compléter mon expérience en apportant une critique chaleureuse mais sans concession.

Je remercie les lecteurs de www.volle.com, dont les remarques m'ont procuré d'innombrables et utiles précisions.

Je remercie l'IEEE et l'ACM pour l'abondante documentation et les revues qu'ils publient (notamment *Communications of the ACM* et *Computer*).

Je remercie les auteurs des ouvrages cités dans la bibliographie, à qui je dois des heures de lecture passionnée et féconde.

Je remercie l'ENSAE, le CEPE, l'ENSPTT, l'Université de Rennes, l'Université Libre de Bruxelles, l'IAE de Paris et l'ENSG qui m'ont procuré un rafraîchissant dialogue avec des étudiants.

Je remercie Laurent Bloch de l'INSERM, Jean-Marie Faure du Crédit Agricole, Jean Kott des Galeries Lafayette, Jean Pavlevski des éditions Economica et Pascal Rivière de l'INSEE, qui m'ont encouragé à composer cet ouvrage.

Laurent Bloch m'a par ailleurs permis de m'approprier à \LaTeX , que j'ai utilisé ici.

Je remercie ceux qui m'ont aidé à corriger les premières versions du texte : Jean-Philippe Carillon, Patrick Declairieux, Bernard Dierickx, David Fayon de la Poste, Jean-Jacques Kasparian de l'INSEE, Thierry Leblond du ministère de la défense, Alain Le Diberder de CLVE, Pierre Musso de l'université de Rennes, Henri Nadel de l'université Paris VII, Lionel Ploquin de la DGI, Raphaël Rousseau de Libroscope, Jean-Claude Serlet, François de Valence et ma sœur Josette Volle dont la lecture soigneuse m'a permis de corriger plusieurs coquilles (mais non pas toutes, je le crains).

* *

Pendant que j'écrivais me sont revenus les souvenirs d'utiles conversations avec plusieurs personnes. J'indique, quand je le connais, le nom de l'entreprise où elles travaillaient alors :

François Andrieux à l'ANPE, Patrick Badillo au CNET, Michel Bernard à Air Inter, Christophe Berthier à Bouygues Telecom, Jean-Michel Beving à SAP, Sylvie Billard à l'ANPE, Christian Blanc et Gilles Bordes-Pagès à Air France, Jean Bouvier à l'université Paris I-Sorbonne, Denis Braleret à Eutelis, Jacques Buisson à Arcome, Jérôme Cabouat au Club des maîtres d'ouvrage, François du Castel au CNET, Ghislaine Clot-Lafleur à l'ENSPTT, Matthieu Colas-Bara à Khiplea, Laurent Collet à la MSA, Isabelle Contini à l'ANPE, Philippe Cottin à la MSA, Jean-Pierre Coudreuse au CNET, Nicolas Curien à France Telecom, François Darbandi à Bouygues Telecom, Lionel David à l'ANPE, Suzanne Debaille à Arcome, Michèle Debonneuil à l'INSEE, Xavier Debonneuil à la Société Générale, Michel Delsaux à Air France, Philippe Desfray à Softeam, Alain Desrosières à l'INSEE, Gérard Dubois au CNET, Jean-Claude Dupoty à Air France, Philippe Estèbe, Hervé Facci à Eutelis, Abdelfatteh Fakhfakh à la Banque Internationale Arabe de Tunisie, Martine Fiévet à l'ANPE, Jean-Paul Figer à Cap Gemini, Michel Frybourg à l'Académie des Technologies, Michel Garcin à la French American Foundation, Édouard Gaulué à l'ENSG, Alain Fournier à Incomsat, Roger Gauthier à Air France, Jacob Genelle à l'ENSG, Michel Gensollen à France Telecom, Yann Gourvenec à France Telecom, Bernard Guibert à l'INSEE, Jacques Guichard, Bernard Hennion et Dominique Henriot au CNET, Jean-Pierre Huin à CENT, Francis Jacq à Eutelis, Jean-Marc Jancovici à X-Environnement, Gérard Jean à Altime, Jean Joskowicz à l'AFISI, Yannick Jouannin à Nomia, Dimitri Kannounikoff à l'ENSG, Jean Laganier à l'INSEE, Antoine Laurès à Arcome, René Lefebvre à l'ANPE, Mallory Lejemble à l'ACOSS, Hervé Lereau à France Telecom, Lionel Levasseur au CNET, Michel-Louis Lévy à l'INED, Witold Litwin à l'université Paris-Dauphine, Christophe Longépé à la Société Générale, Edmond Malinvaud à l'INSEE, René Mandel à Oresys, Michel Mangonaux à l'ANPE, Marie Marchand à France Telecom, Alain de Mijolla à l'AFPA, Jacky Noviant à Air France, Didier Ott à Bouygues Telecom, Jean-Louis Peaucelle à l'université de la Réunion, Philippe Penny à Eutelis, Diane de Pierrefeu au club des maîtres d'ouvrage, Philippe Plazanet à la Banque Indosuez, Alex Pringault à Lagardère Active, Jacques Printz au CNAM, Yvon Quérou et Michel Ram-

bourdin à Eutelis, Louis Réchaussat à l'INSERM, Olivier Renaud à Softeam, Claude Rochet au CIGREF, Jean Rohmer à Idéliance, Jean-François Sauer à AXA, Jean-Cyril Spinetta à Air France, Christophe Talière à Eutelis, Patrick Teisserenc à l'État-Major de l'Armée de Terre, Michèle Thonnet au ministère de la santé, Claude Truffier, Paul Vidal et Julien Varin à l'ANPE, Serge Yablonsky à l'AFAI, Jean Zeitoun à MOST.

Que ceux qui m'ont aidé par leurs remarques et leurs critiques, ou dont j'ai consulté la documentation sur la Toile mais dont le nom ne figure pas ci-dessus veuillent bien me pardonner ma mauvaise mémoire et qu'ils soient assurés de ma reconnaissance.

Première partie

Comment l'Automate
Programmable doué
d'Ubiquité assiste l'Être
Humain Organisé

Chapitre 1

Du côté de l'ordinateur

« Ce qui est unique dans l'ordinateur (...) c'est sa baisse de prix exponentielle ininterrompue depuis trente ans. En divisant par 1000 le coût d'une même opération, la machine s'est ouvert sans cesse de nouvelles applications. En ce sens, elle bouleverse la société. (...) Les coûts des circuits individuels baissent (...) de 20 à 30 % chaque année, à niveau égal de performance. (...) Il y a là un phénomène unique par son ampleur et sa durée, et qui explique à lui seul la formidable croissance de l'informatique » (Jean-Pierre Brulé, [28] p. 61-62)

De tous les outils de l'informatique, l'ordinateur est celui qui nous est le plus familier. Mais est-il bien nommé? Non; le mot « ordinateur » est un de ces « faux amis » qui abondent dans le vocabulaire de l'informatique. S'il ne présente aucun inconvénient pour l'informaticien qui sait exactement ce que ce mot recouvre, il provoque des contresens dans le public et chez les personnes cultivées mais inexpertes qui s'efforcent de comprendre l'informatique.

Il serait dérisoire de tenter de déraciner un mot que l'usage a consacré, mais nous proposerons ci-dessous, pour redresser le faisceau de ses connotations, de lui associer mentalement l'expression « automate programmable ».

Un *automate*, c'est une machine qui accomplit exactement, et dans l'ordre, les opérations pour lesquelles elle a été conçue. La liste de ces opérations n'est pas nécessairement écrite sous la forme d'un programme: elle peut résulter de l'enchaînement d'une série d'actions mécaniques. Le « canard digérateur » de Vaucanson (1739) savait picorer des grains de maïs, les broyer, les mêler à de l'eau et les rejeter; il imitait ainsi le vrai canard qui mange et rejette des excréments sans bien sûr lui ressembler en rien du point de vue de l'anatomie. Le métier Jacquard (1801) est un automate qui obéit à un *programme* inscrit sur un carton perforé, mais il ne sait accomplir qu'un seul type d'opération: le tissage.

Il a fallu un étonnant effort d'abstraction pour oser mettre entre parenthèses toute application possible et concevoir l'*automate pur et absolu*, construit pour obéir à tout type de programme et commander à d'autres

machines l'exécution des opérations les plus diverses (hauts parleurs, écrans et imprimantes de l'ordinateur ; bras articulés des robots ; ailerons des avions en pilotage automatique ; commande des moteurs, suspension et freins des automobiles etc.).

Cet automate absolu, c'est l'ordinateur. Il est *essentiellement* programmable ; on peut l'utiliser pour faire du traitement de texte, du dessin, du calcul, de la musique, il est incorporé dans les équipements électromécaniques les plus divers. Le programme se substitue, de façon économiquement efficace, aux engrenages et ressorts auparavant nécessaires pour commander mécaniquement l'exécution d'une série d'actions.

* *

L'image que nous avons aujourd'hui de l'ordinateur est datée. Elle ne correspond ni à ce qu'il était dans les années 1950, 60 et 70, ni à ce qu'il sera dans dix à quinze ans. Beaucoup de nos objets familiers (téléphone mobile, « Palmtop », carte à puce etc.) sont des ordinateurs sans que nous nous en soyons avisés. Les ressources de mémoire et de puissance auxquelles notre écran-clavier donne accès ne sont pas seulement celles qui se trouvent sur notre machine : *via* le réseau (Ethernet, Intranet ou Internet) l'ordinateur a acquis l'*ubiquité* : la localisation de ses ressources physiques est indifférente. Ainsi nous n'utilisons pas *des* ordinateurs (chacun le sien) mais nous partageons, dans la limite de nos droits d'accès et habilitations, *un* ordinateur, *la* machine constituée de réseaux, mémoires, processeurs et programmes, l'« automate programmable doué d'ubiquité ».

Pour comprendre cet être devenu banal mais qui reste d'une grande complexité, il faut articuler les logiques qu'il met à l'œuvre : un « modèle en couches » sera ici précieux. Il faut aussi situer la racine du phénomène de l'informatisation : l'informatique apporte au rapport entre l'être humain et la nature un changement analogue à celui qui aurait résulté de la découverte d'une nouvelle ressource naturelle. L'être humain, qu'on le considère dans la vie sociale ou dans l'entreprise où il s'organise pour produire, tire parti de la synergie entre les propriétés électroniques des semi-conducteurs et les propriétés logiques du dispositif de commande de l'automate, le « langage de programmation ».

Cette mise en exploitation a nécessité la maîtrise de certains procédés techniques que l'on a malencontreusement nommées « nouvelles technologies ». Le progrès des performances, certes rapide, s'est donc étalé dans le temps (« loi de Moore »). Il a suscité une baisse de prix, également rapide, qui a favorisé la pénétration universelle de l'ordinateur dans la société comme dans les entreprises.

1.1 Un changement du rapport avec la nature

On peut représenter l'effet des TIC¹ sur l'économie selon un modèle à trois couches (figure 1.1) :

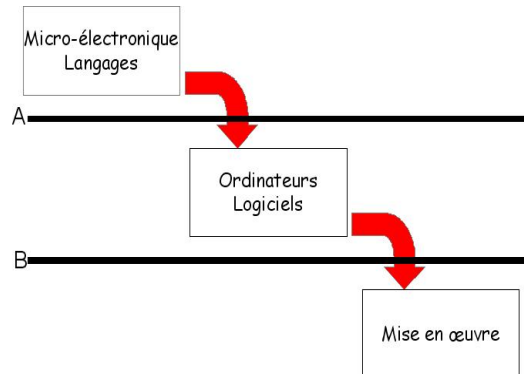


FIG. 1.1 – Cascade des TIC

- à la source se trouvent deux *techniques fondamentales* : d'une part la fabrication des microprocesseurs et mémoires ; d'autre part les systèmes d'exploitation, langages et outils de programmation. Ces deux sous-ensembles sont d'ailleurs reliés entre eux : on n'utilise pas les mêmes langages de programmation selon la ressource de mémoire ou de puissance disponible ;

- immédiatement en aval de cette source se trouvent les *équipements* qui mettent en œuvre les techniques fondamentales (ordinateurs, réseaux, terminaux etc.), ainsi que les *logiciels* ;

- en aval des équipements et logiciels se trouve enfin leur *mise en œuvre* par les entreprises, associée à la maîtrise des processus de production, à la définition de la relation avec les clients, fournisseurs et partenaires, ainsi qu'à des formes spécifiques de concurrence.

* *

Pour étudier les effets des TIC sur l'économie on doit situer la frontière entre ce qui est appelé « TIC » et ce qui est appelé « reste de l'économie ». La solution la plus courante consiste à considérer que les ordinateurs et autres machines utilisant les techniques fondamentales, ainsi que les logiciels, relèvent des TIC. Alors la frontière se situe au niveau B de la figure 1.1.

Certes, personne ne peut nier que les ordinateurs, commutateurs etc. ne soient des représentants éminents des TIC. Cependant les évolutions de ces machines résultent, pour l'essentiel, des progrès réalisés dans les techniques

1. « Techniques de l'information et de la communication ». Dans les années 1990 on disait NTIC, « Nouvelles technologies de l'information et de la communication ». L'adjectif « nouvelles » n'est plus de mise aujourd'hui et « technique » est plus exact que « technologie », qui signifie « discours sur la technique ».

fondamentales. Par exemple la croissance des performances des microprocesseurs et mémoires dont la « loi de Moore » rend compte est déterminante pour l'évolution des ordinateurs.

Dans la couche finale, celle des utilisations, il s'agit de tirer le meilleur parti des évolutions que permettent les ordinateurs, réseaux etc. ; dans la couche intermédiaire, il s'agit de tirer le meilleur parti des ressources offertes par les techniques fondamentales. Si chacune de ces deux couches obéit à une logique qui lui est propre, le moteur de leur évolution se trouve en amont. Si l'on souhaite isoler la *source* de l'évolution, qui réside dans les techniques fondamentales, il faut donc placer la frontière au niveau A.

* *

Dans la couche initiale, celle des techniques fondamentales, il ne s'agit pas d'utiliser des ressources produites en amont, mais de *créer* des ressources par la maîtrise des propriétés *physiques* du silicium et (osons le dire) par la maîtrise des conditions *mentales* de la production et de l'utilisation des langages de programmation, le terme « mental » désignant ici un ensemble de dimensions intellectuelles, psychologiques et sociologiques.

Ainsi, alors que les deux autres couches résolvent un problème *économique* (il s'agit de *faire au mieux avec les ressources dont on dispose*), la couche initiale considère *la nature elle-même*, sous les deux aspects de la physique du silicium et de la « matière grise » des êtres humains, aspects dont elle vise à faire fructifier la synergie. Dans la couche des techniques fondamentales s'opère donc un *changement du rapport avec la nature* ; dans les deux autres couches s'opère *l'adaptation à ce changement*.

Élargir, par des procédés de mieux en mieux conçus, les ressources que fournit la nature, c'est une tâche analogue à la découverte ou plutôt à l'exploration progressive d'un continent nouveau que des pionniers transformeraient et équiperait pour produire des biens utiles. Or découvrir un continent, puis l'explorer pour le mettre en exploitation, c'est transformer les prémisses de la réflexion et de l'action économiques, c'est modifier les conditions de la vie en société.

* *

Tout raisonnement économique s'appuie sur des *exogènes* : il suppose données les techniques, ressources naturelles, fonctions d'utilité et dotations initiales. Il en tire les conséquences, élucide les conditions de leur utilisation optimale, mais ne tente pas d'expliquer leur origine. Sans doute la recherche du profit n'est pas pour rien dans l'ardeur des pionniers ou des chercheurs ; mais cette ardeur se dépenserait en pure perte si elle ne pouvait pas mettre en exploitation une ressource naturelle féconde (ici le silicium, la « matière grise », et leur synergie).

Le *changement du rapport entre les êtres humains et la nature* que l'on rencontre donc dans les techniques fondamentales n'est ni économique, ni sociologique, même s'il a des conséquences économiques et sociales. L'innovation qui se déverse dans l'économie, dans la société, à partir des techniques fondamentales est analogue à un phénomène naturel extérieur à l'action humaine qu'il conditionne comme le font le climat, les

courants océaniques, la reproduction des êtres vivants, les gisements que nous a légués l'histoire géologique de la Terre etc.

Est-ce à dire que l'économie ou la sociologie n'ont rien à voir avec les TIC? certes non, puisqu'elles doivent répondre aux problèmes que pose leur bonne utilisation: les exogènes étant modifiées, comment « faire au mieux avec ce que l'on a² » et qui est nouveau? comment faire évoluer des institutions qui s'étaient lentement adaptées aux exogènes antérieures, mais qui ne le sont pas nécessairement aux exogènes nouvelles? comment définir le « savoir-vivre » nouveau qui permettra de préserver la cohésion sociale dans une société que bouleverse l'innovation?

Cette tâche n'est pas facile. Considérons les efforts que doivent réaliser les entreprises pour modifier leurs processus de production et les conditions de travail des agents opérationnels; pour adapter les périmètres des directions, les missions et les espaces de légitimité des dirigeants; pour équiper et faire évoluer les relations avec les clients, partenaires et fournisseurs etc.

1.2 Qu'est-ce qu'un ordinateur?

Si l'on vous dit « qu'est-ce qu'un ordinateur? », une image semblable à celle de la figure 1.2 vous viendra sans doute à l'esprit.



FIG. 1.2 – *L'image aujourd'hui la plus courante de l'ordinateur*

C'est ainsi en effet que se présente aujourd'hui l'ordinateur sur la plupart des bureaux (*desktop*): un écran, un clavier, une souris, une unité centrale. Ajoutons ce que le dessin ne montre pas mais qui est essentiel: un modem connecté à une prise téléphonique, ou une carte Ethernet connectée à un

2. Cette phrase résume l'ambition pratique de la science économique, dont on peut dire qu'elle n'est rien d'autre qu'une *théorie de l'efficacité*.

réseau local. L'ordinateur en réseau équipe aujourd'hui la quasi totalité des postes de travail dans l'entreprise, et nombre de logements.

Une autre image rivalise avec la précédente : celle de l'ordinateur portable (*laptop*, on le pose sur les genoux) composé d'un boîtier plat dépliant incorporant écran, clavier et souris (figure 1.3).



FIG. 1.3 – Autre image courante

L'ordinateur portable coûte 50 % de plus qu'un ordinateur de bureau, il n'a ni les mêmes performances ni la même fiabilité, mais il est commode pour les personnes qui doivent se déplacer souvent et ont besoin d'emmener leur ordinateur avec elles.

* *

Ces deux « incarnations » de l'ordinateur sont datées. L'ENIAC (*Electronic Numerical Integrator and Computer*) ne ressemblait en rien aux ordinateurs ci-dessus (figure 1.4).³

Les ordinateurs sont restés de très grosses machines jusqu'aux années 1960. Le remplacement des lampes par des circuits intégrés dans les années 1950 permit de réduire leur taille alors que leurs performances augmentaient, mais ils occupaient encore beaucoup de place et avaient besoin d'un local climatisé. Ils n'avaient ni écran, ni souris, ni clavier (si ce n'est celui du télétype de l'opérateur) : les commandes étaient perforées sur des cartes, les résultats imprimés sur des *listings*.

Ceux qui ont débuté en informatique dans les années 1960 se rappellent les paquets de cartes que l'on faisait passer par un guichet vers les opérateurs et qui revenaient, un ou deux jours après, accompagnés d'un listing ; celui-ci contenait une liste d'erreurs qu'il fallait corriger avant de faire passer de nouveau le paquet de cartes par le guichet jusqu'à convergence du processus. En 1968, on dénombrait 30 000 ordinateurs dans le monde.

3. L'ENIAC n'a pas été le premier ordinateur (Bloch [14]) : comme il fallait une intervention manuelle pour le préparer à la réalisation d'un calcul, il n'était donc ni véritablement programmable ni conforme à l'architecture de von Neumann [146]. Les deux premiers ordinateurs au sens plein du terme furent britanniques : en 1948 le MARK I, réalisé sous la direction de Max Newman à l'université de Manchester, et en 1949 l'EDSAC de Maurice Wilkes à l'université de Cambridge.

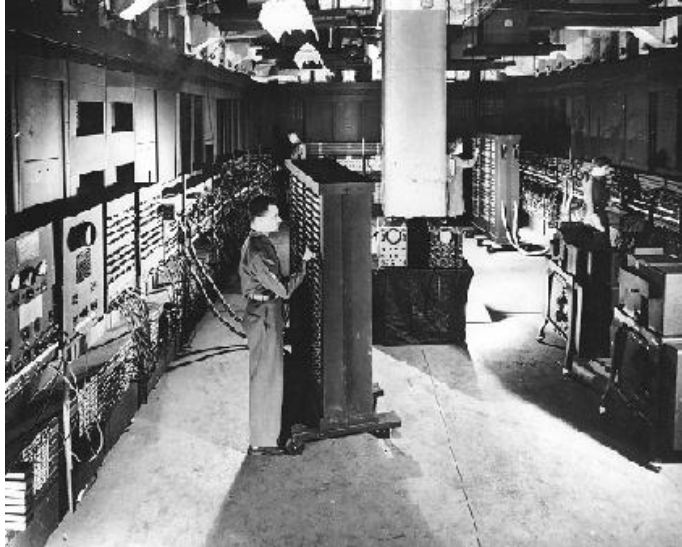


FIG. 1.4 – L'ENIAC, 4 février 1946

Dans les années 1970 se sont mis en place des terminaux permettant d'accéder aux *mainframes* en temps partagé. Sur ces couples écran-clavier on pouvait composer et tester les programmes en « temps réel », ce qui accélérât notablement la production. Certes leurs écrans noirs et leurs caractères verts étaient austères, mais ils représentaient un grand progrès par rapport aux bacs à cartes.

La conquête de l'autonomie de l'utilisateur n'est venue qu'avec le micro-ordinateur : les premiers sur les bureaux furent l'Apple II (1977) puis le PC d'IBM (1981) et le Macintosh (1984).

Ingénierie de l'ordinateur

L'ordinateur est un automate programmable, donc adaptable à toutes les tâches qu'on lui assigne. En dehors du processeur et de la mémoire, ses autres organes sont comme des bras, des mains, des capteurs sensoriels. L'ingénierie de l'ordinateur est la plus compliquée que l'être humain ait jamais conçue. Elle est compliquée non seulement dans les couches physiques (processeurs, mémoires, câblages) où s'organise la circulation des électrons, se régulent les tensions électriques, se dessinent les bits et se réalisent en binaire les opérations de consultation des mémoires, calcul et écriture, mais aussi dans l'empilage des diverses couches de langage qui sont nécessaires pour commander l'automate : microcode, assembleur, système d'exploitation, langages de programmation, applications.

De quoi a besoin l'ordinateur pour fonctionner? D'une mémoire, d'un processeur, d'un système d'exploitation et de programmes. Toute machine munie d'un processeur et d'une mémoire devient un ordinateur dès que l'on y charge des programmes.

Architecture de l'ordinateur

Dans le monde des programmeurs, le mot « architecture » désigne le jeu d'instructions à partir duquel il sera possible de programmer des compilateurs ou des interpréteurs et donc de définir des langages de programmation : on parle ainsi de l'architecture RISC (*Reduced Instruction Set Computer*), des architectures 80x86 (pour désigner les microprocesseurs 8086, 80386 et 80486 d'Intel^a) etc. Les autres choix que doit faire l'architecte qui conçoit un ordinateur sont alors désignés par le terme « implémentation ».

Hennessy et Patterson ([153] p. 9) estiment cependant qu'il faut donner au mot « architecture » une acception plus large, recouvrant l'ensemble des choix que fait l'architecte. Outre le jeu d'instructions, qui est la vue que l'ordinateur présente au programmeur, ces choix concernent l'*organisation* (système mémoire, structure de bus, caches, pipelines etc.) et le *matériel* (fréquence d'horloge etc.) : un même jeu d'instruction peut en effet être offert par des processeurs résultant de choix d'organisation et de matériel différents.

On distingue parmi les ordinateurs trois grandes familles fonctionnelles à chacune desquelles correspond un type d'architecture spécifique :

- les ordinateurs de bureau demandent un microprocesseur de pointe, intégrant le graphique et les systèmes d'entrée et sortie ;
- pour les serveurs, il s'agit d'intégrer plusieurs microprocesseurs, souvent dans un multiprocesseur, et de les accompagner d'entrées et sorties extensibles ;
- pour les ordinateurs embarqués, il faut tirer parti de processeurs haut de gamme tout en réduisant le coût et la consommation d'énergie.

^a Certains estiment qu'il est « difficile d'expliquer et impossible d'aimer » l'architecture 80x86, à laquelle ils préfèrent l'architecture RISC (Hennessy et Patterson [153] p. 1061 et Wilkes [218]).

L'image de l'ordinateur que nous venons d'évoquer est donc trop étroite. Il convient de ranger sous le concept d'ordinateur les commutateurs du réseau téléphonique, nombre de nos appareils ménagers (ceux qui comportent une mémoire, un processeur et des programmes), nos avions, nos automobiles etc. ou du moins la partie de ces équipements qui assure l'exécution de programmes informatiques et que l'on appelle « ordinateur de bord » ou « ordinateur embarqué ». Méritent également le nom d'ordinateur nos *Palmtops* (on les tient dans la paume de la main), téléphones mobiles et cartes à puce (figure 1.5).



FIG. 1.5 – Des ordinateurs qui n'ont pas l'air d'être des ordinateurs

Des recherches sont en cours pour accroître encore la portabilité de l'ordinateur : le *wearable computer* est « portable » au sens où l'on dit que l'on porte des vêtements. Même si les prototypes sont quelque peu monstrueux, il suffit d'extrapoler leur miniaturisation pour voir qu'ils ne seront bientôt pas plus encombrants qu'une paire de lunettes et un téléphone mobile (figure 1.6). Un appareil comme le Treo de Handspring conjugue déjà les fonctionnalités du téléphone mobile et certaines de celles de l'ordinateur (figure 1.7).

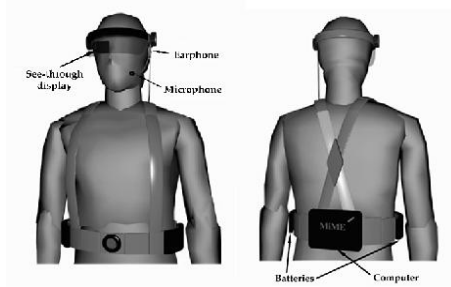


FIG. 1.6 – Le « *Wearable computer* »

Connecté en permanence par une liaison sans fil à haut débit, le *wearable* permettra à chacun d'accéder, où qu'il soit, à une ressource informatique personnelle résidant sur des serveurs dont la localisation lui sera indifférente. L'ubiquité de l'ordinateur ne sera plus alors conditionnée par la proximité entre l'utilisateur et un poste de travail : *c'est le corps de l'utilisateur lui-même qui sera équipé, et non plus son bureau*. L'écran sera remplacé par exemple par des lunettes sur lesquelles se formera l'image ; le clavier, par un cahier de touches dépliant, par des touches projetées sur une table ou par la reconnaissance vocale. Cet équipement fournira une prothèse à la mémoire comme à la recherche d'information. On anticipe sa puissance, les risques d'accoutumance qu'il comportera et les obstacles qu'il opposera aux rapports humains. Nous devons, nous devons déjà apprendre à maîtriser l'ordinateur pour qu'il ne nous dévore pas⁴.

4. Mais avons-nous maîtrisé en France l'automobile, qui en 2000 a tué 7 600 personnes



FIG. 1.7 – Vers la fusion du téléphone mobile et de l'ordinateur

Dans une dizaine d'années, l'image que tout le monde se fait aujourd'hui de l'ordinateur sera obsolète, tout comme sont aujourd'hui obsolètes les images de l'ENIAC, des cartes perforées, des grappes de terminaux reliées à un *mainframe*, ou pour remonter plus loin dans le temps les rangées de bureaux équipés de machines à calculer que l'on voyait encore dans les entreprises à la fin des années cinquante⁵. Au delà de ces images éphémères, le raisonnement réclame la fermeté d'un concept pérenne.

1.3 Qualité du vocabulaire

Quelle est la qualité descriptive et explicative du concept d'« ordinateur » ? Correspond-il à la réalité à laquelle nous confronte l'histoire de l'informatique ?

À tout concept est attaché un mot ; à tout mot sont attachés d'une part l'image centrale qui sert de pivot au concept et d'autre part aussi, dans l'usage courant, un faisceau de connotations qui lui associe d'autres concepts et d'autres images.

Lorsqu'on examine un concept, on doit donc d'abord se demander si le pivot est bien placé, si l'image qu'évoque le mot correspond à la réalité qu'il s'agit de décrire ; puis on doit se demander si les connotations, les associations d'idées que le mot suggère sont de nature à enrichir sa compréhension ou à égarer l'imagination sur de fausses pistes.

Les disciplines intellectuelles mettent beaucoup de temps à mûrir, à construire leur vocabulaire et leurs méthodes. L'informatique, ayant à peine plus de cinquante ans, est une discipline très jeune. Les spécialistes passionnés qui l'ont construite ont dû faire flèche de tout bois : peu leur importait le choix des mots car, comme le disait Hilbert, une fois que l'on sait de quoi

et en a blessé 162 100 (source : ONISR) ? Le tabac, qui tue chaque année plus de 50 000 personnes ? La télévision, qui accapare notre attention 3 heures 20 par jour en moyenne (source : Médiamétrie) ? La consommation d'énergie, qui dégrade le climat (Jancovici [90]) ? L'ordinateur n'est ni le seul défi posé à la sagesse humaine, ni sans doute le plus grave, même si la maturation de ses usages pose des questions délicates.

5. On en voit un exemple dans le film *The Apartment* de Billy Wilder (1960).

l'on parle peu importe si l'on dit « chaise » et « table » au lieu de « droite » et « point⁶ ».

Le vocabulaire de l'informatique a été ainsi frappé d'une sorte de malédiction : presque tous les termes qui le composent sont de faux amis au sens où l'on dit qu'un mot d'une langue étrangère a un « faux ami » en français, un mot qui lui ressemble mais qui n'a pas le même sens⁷. Un phénomène aussi général ne peut pas être dû au seul hasard : soit ces termes datent d'une époque révolue et font référence à des usages qui n'ont plus cours ; soit leurs inventeurs avaient des idées confuses, ont été maladroits, ou encore ils ont délibérément cherché à créer du désordre.

Dans le vocabulaire de l'informatique le mot « informatique » lui-même est un des rares qui soient sans reproche : malheureusement l'usage n'a pas tiré le meilleur parti de son potentiel sémantique (voir page 198). Le mot « logiciel », lui aussi bien trouvé⁸, est supérieur à l'anglais « software » qu'il traduit⁹. Mais « ordinateur » est un faux ami tout comme le sont « langage », « objet », « numérique », « donnée » et « information » ; en anglais, « computer » est lui aussi un faux ami.

L'usage dictant sa loi nous utiliserons ces termes, mais nous aurons soin de remplacer mentalement leurs connotations malencontreuses par d'autres plus exactes. Passons-les en revue.

1.3.1 « Ordinateur »

L'anglais « computer » signifie « calculateur »¹⁰. Ce mot représente-t-il convenablement le concept d'« ordinateur » ? Non, car lorsque nous utilisons l'ordinateur pour faire du traitement de texte, du dessin, ou encore pour consulter la Toile, les opérations qu'il exécute ne relèvent pas du calcul même si elles sont comme on dit « numérisées ». La dénomination « computer » correspondait à la mission de l'ENIAC (calculer des tables pour aider les artilleurs à régler leurs tirs) mais elle ne correspond pas à celle des ordinateurs d'aujourd'hui.

En 1954 IBM voulait trouver un nom français pour ses machines et éviter le mot « calculateur » qui lui semblait mauvais pour son image. Le

6. « Man muß stets statt „Punkt“, „Gerade“, „Ebene“ auch „Tisch“, „Stuhl“, „Bierseidel“ sagen können » (« Il faut toujours pouvoir dire “table”, “chaise” et “bock de bière” à la place de “point”, “droite” et “plan” », David Hilbert, 1862-1943).

7. C'est le cas par exemple du mot « virtuel ». En anglais, « virtual » signifie « being such in essence or effect though not formally recognized or admitted (a virtual dictator) » (*Merriam Webster's College Dictionary*). En français, il signifie exactement le contraire : « qui est seulement en puissance et sans effet actuel » (Littré). Il en résulte de pénibles contresens lorsqu'on parle de « circuit virtuel » en télécommunications, d'« entreprise virtuelle » en économie, d'« espace virtuel » en informatique etc.

8. « Logiciel » est une création administrative réussie. Il a été introduit par l'arrêté du 22 décembre 1981 relatif à l'enrichissement du vocabulaire de l'informatique (*Journal officiel* du 17 janvier 1982).

9. « Software » a été inventé pour faire contraste avec « hardware », qui désigne le matériel.

10. « Computer » vient du latin *cumputare*, « compter », lui-même composé de *cum*, avec, et *putare*, penser.

Computer: « A programmable electronic device that can store, retrieve, and process data » (*Merriam Webster's Collegiate Dictionary*)
 « A general-purpose machine that processes data according to a set of instructions that are stored internally either temporarily or permanently. The computer and all equipment attached to it are called “hardware”. The instructions that tell it what to do are called “software”. A set of instructions that perform a particular task is called a “program” or “software program”. »

linguiste Jacques Perret proposa, dans sa lettre du 16 avril 1955, d'utiliser « ordinateur¹¹ », mot ancien passé d'usage qui signifiait « celui qui met en ordre » et désigne en liturgie celui qui confère un ordre sacré.

« Ordinateur » est élégant mais c'est un faux ami plus dangereux encore que « computer ». L'ordinateur met-il vos affaires en ordre ? Certes non. C'est *vous* qui devez veiller à les tenir en ordre : si vous n'y prenez pas garde, un désordre inouï se créera dans vos dossiers électroniques. L'ordre ne peut venir que de l'opérateur humain, non de la machine.

Ordinateur: « Machine capable d'effectuer automatiquement des opérations arithmétiques et logiques (à des fins scientifiques, administratives, comptables etc.) à partir de programmes définissant la séquence de ces opérations » (*Dictionnaire Hachette*).
 « Machines automatiques de traitement de l'information permettant de conserver, d'élaborer et de restituer des données sans intervention humaine en effectuant sous le contrôle de programmes enregistrés des opérations arithmétiques et logiques. » (*Quid*)

Il ressort des définitions usuelles que l'ordinateur, c'est un « automate programmable ». Pour indiquer que cet automate est accessible depuis n'importe quel poste de travail en réseau, il faut ajouter l'adjectif « doué d'ubiquité¹² ». L'« ordinateur », c'est un « automate programmable doué d'ubiquité », un « APU ».

Dans une entreprise ce singulier désigne non chaque machine isolément (le serveur central, le poste de travail, les routeurs etc.), mais l'ensemble technique, logique et fonctionnel que constituent ces machines et qui est mis à la disposition de l'utilisateur sous la seule contrainte de ses habilitations.

11. Ce néologisme n'a pas connu un grand succès dans les autres langues : si l'on dit « ordenador » en espagnol, on dit « Computer » en allemand, « calcolatore » en italien, « computador » en portugais, « компьютер » en russe. Le chinois utilise deux idéogrammes qui signifient « cerveau électrique », le japonais transcrit phonétiquement le mot « computer ».

12. On pourrait dire « omniprésent », mais cet adjectif n'a pas exactement le même sens que « doué d'ubiquité ». En anglais on peut utiliser « ubiquitous ». Le « computer », c'est un *Ubiquitous Programmable Automat*, un « UPA ».

Lorsque nous sommes devant un poste de travail, les ressources de puissance et de mémoire dont nous disposons ne sont pas en effet seulement celles qui se trouvent dans le processeur, la RAM ou le disque dur de cette machine¹³, mais aussi celles auxquelles le réseau nous donne accès : c'est cet ensemble que nous appellerons « ordinateur », au singulier. Lorsque cet ensemble est en panne, on entend dire « l'informatique est en panne » ou « mon ordinateur est en panne ».

La diversification que procure à l'ordinateur son caractère programmable ne doit pas faire oublier qu'il s'agit d'un automate : il exécute les instructions dans l'ordre où elles lui ont été données et, contrairement à l'être humain, *il est insensible aux connotations*. Cela lui confère à la fois une grande précision¹⁴ et une extrême raideur. Pour comprendre ce qui se passe d'une part dans la tête du programmeur, d'autre part dans le processeur de l'automate, il faut avoir fait l'expérience de la programmation ; à défaut on peut lire l'excellent petit livre *Karel the Robot* (Pattis, [154] ; voir page 32).

1.3.2 « Langage »

« The logical mind-frame required for programming spilled over into more commonplace activities. You could ask a hacker a question and sense his mental accumulator processing bits until he came up with a precise answer to the question you asked. Marge Saunders would drive to Safeway every Saturday morning in the Volkswagen and upon her return ask her husband, "Would you like to help me bring in the groceries?" Bob Saunders would reply, "No". Stunned, Marge would drag in the groceries herself. After the same thing occurred a few times, she exploded, hurling curses at him and demanding to know why he said no to her question. "That's a stupid question to ask", he said. "Of course I won't like to help you bring in the groceries. If you ask me if I'll help you bring them in, that's another matter." It was as if Marge had submitted a program into the TX-0, and the program, as programs do when the syntax is improper, had crashed. It was not until she debugged her question that Bob Saunders would allow it to run successfully on his own mental computer. » (Levy, [113] p. 37-38).

On utilise en informatique le mot « langage » pour désigner la liste des instructions et les règles d'écriture qui permettent de composer un programme pour ordinateur. Ce que l'on appelle ici « langage », c'est donc *le dispositif de commande de l'automate*.

13. La RAM (« Random Access Memory ») est la mémoire sur laquelle travaille la machine. Son accès est rapide (« random » signifie que le délai d'accès est le même quel que soit l'emplacement de la donnée dans la RAM). Le « disque dur » est une mémoire de masse à accès lent ; contrairement à la RAM il conserve les données lorsque l'ordinateur s'éteint.

14. Toutefois cette précision a des limites, voir page 83.

Rapports entre le programmeur et l'ordinateur

Karel the Robot (Pattis [154]) fournit une utile métaphore de la programmation. On part d'un jeu : il s'agit de commander un robot nommé Karel qui se déplace dans un monde simple. Le plan de ce monde est un quadrillage semblable aux rues d'une ville américaine ; Karel peut s'y déplacer en avançant d'un carré et en tournant d'un quart de tour à droite (en répétant les quarts de tour il peut faire des virages ou des demi-tours). Le chemin lui est parfois barré par un mur qu'il ne perçoit que lorsqu'il se trouve juste devant. Il porte enfin un sac contenant des balises qu'il peut déposer ou ramasser à certains carrefours.

Karel obéit exactement aux ordres qu'on lui donne. Si on lui donne un ordre impossible (avancer dans un mur, poser une balise alors que son sac est vide), il envoie un message et s'arrête. Bref : Karel est infiniment travailleur et patient, jamais rebuté par une tâche répétitive, mais il ne fait que ce qu'on lui ordonne et ne peut prendre aucune décision. Celui qui programme Karel dispose, lui, d'un langage de programmation.

L'auteur invite à programmer Karel. Il s'agit d'abord d'accomplir des tâches simples (parcourir la diagonale entre deux points, longer un rectangle entouré d'un mur etc.). Puis on écrit des programmes un peu plus difficiles : faire par exemple sortir Karel d'une « pièce » rectangulaire entourée de murs percés d'une porte, quels que soient la forme de la pièce, l'emplacement de la porte et l'emplacement initial de Karel. Pour traiter tous les cas particuliers en un seul programme il faut décomposer des tâches complexes en tâches élémentaires : nous voici dans la programmation structurée, comme en Pascal.

En lisant ce livre on s'habitue à la coopération entre le programmeur humain, avec sa créativité, et un robot d'une patience inlassable. L'intuition découvre le langage qu'il convient de parler à l'ordinateur si l'on veut qu'il obéisse : on apprend à la fois à concevoir un tel langage et à l'utiliser.

Cela permet d'entrevoir les possibilités ouvertes à l'« être humain assisté par ordinateur », concept plus fécond que celui d'intelligence artificielle.

Il existe une différence importante entre un tel « langage » et les langages qui nous servent, à nous êtres humains, pour formuler ou communiquer notre pensée. Un texte énoncé ou écrit par un être humain est fait pour être compris par celui qui le reçoit ; il s'appuie sur les « connotations », ces diffractions de sens secondaires qui entourent chaque mot et qui confèrent au texte une profondeur, un « plein » qui dépasse le sens littéral des mots qu'il contient.

Par contre *un programme est fait pour être exécuté par un automate* et non pour être lu ni compris par un être humain¹⁵ : il sera souvent incompréhensible, même (après quelques jours) pour celui qui l'a écrit. Les expressions qu'il contient ont toutes un sens et un seul, car l'automate ne sait pas interpréter les connotations et ne peut exécuter que des instructions non ambiguës.

Il est vrai que les êtres humains, lorsqu'ils préparent une action, doivent utiliser eux-mêmes un langage purement conceptuel et éviter les connotations : l'ingénierie, la guerre, la science, utilisent des textes aussi « secs » (et parfois aussi incompréhensibles à première vue) qu'un programme informatique. Le mathématicien qui relit une de ses propres démonstrations après quelques mois a autant de mal à la comprendre que le programmeur qui relit un de ses programmes. Cependant, même technique, le langage humain est encore fait pour être entendu par des êtres humains et non pour être exécuté par un automate : les mathématiciens ont recours pour faciliter la lecture à des « abus de langage » qui court-circuitent certaines étapes, jugées évidentes, du raisonnement (Bourbaki [20]). Par contre un programme informatique doit être parfaitement explicite.

On a pu utiliser les mots « grammaire », « syntaxe » et « vocabulaire » pour désigner la structure et les composants d'un langage de programmation ; ces mots sont ici à leur place : les règles formelles de la programmation s'expriment d'une façon analogue à celles du langage humain. Mais cette analogie n'est pas une identité car le langage humain, lui, ne se réduit pas à un formalisme. Parler de « langage » pour désigner le dispositif de commande de l'automate, c'est donc... un abus de langage qui suscite la confusion, notamment dans les réflexions sur l'intelligence de l'ordinateur (voir page 86).

* *

Le langage conceptuel est nécessaire à l'action : pour agir efficacement dans le monde de la nature¹⁶, il importe de désigner les objets avec une parfaite *précision*. Par contre, dans la phase exploratoire qui précède l'action

15. Les théoriciens de l'informatique disent qu'un informaticien doit savoir lire les programmes, mais c'est là un abus de langage. Ils devraient plutôt dire qu'il faut savoir *déchiffrer* les programmes : « It is exceedingly important to acquire skill in reading other people's computer programs, yet such training has been sadly neglected in too many computer courses and it has led to some horribly inefficient uses of computing machinery » (Knuth [105] vol. 1, p. 170).

16. Nous serons amenés, pour analyser le phénomène de l'informatisation, à articuler selon un modèle en couches le « monde de la nature », le « monde de la pensée » et le « monde des valeurs » : voir page 212.

et la construction conceptuelle, nous procédons par analogie, association d'idées, et relierons par des *connotations* les divers domaines de l'expérience.

Le langage connoté est l'humus sur lequel se forme le langage conceptuel. Sans humus, pas de plante possible - mais l'humus n'est pas lui-même une plante et on ne saurait le manger. De même, sans langage connoté, pas de langage conceptuel et donc pas d'action possible ; mais le langage connoté ne peut pas nourrir directement l'action.

Certaines personnes, attachées à la fécondité du langage connoté et sensibles à la richesse du monde qu'il permet d'évoquer (l'allusion poétique comble les lacunes du langage comme la succession des images crée au cinéma la sensation du mouvement continu) refusent la « sécheresse » du langage conceptuel ; ce faisant elles se mutilent du côté de l'action volontaire et se limitent à un rôle contemplatif. Ce rôle peut sans doute apporter des plaisirs esthétiques, mais non les plaisirs et leçons de l'action.

D'autres personnes, attachées à des finalités pratiques et éprises d'efficacité, refusent au contraire l'ambiguïté du langage connoté et ne veulent utiliser que le langage conceptuel. C'est souvent le cas des ingénieurs et des informaticiens. Ils en viennent à se couper des autres, auxquels ils parlent avec la même rigueur formelle que s'ils écrivaient un programme (voir la citation page 31).

Ainsi les ingénieurs font des reproches à ceux qu'ils qualifient de « littéraires » ou de « poètes » (philosophes, sociologues, historiens et autres), et d'autre part certains sociologues, philosophes etc. exècrent les ingénieurs, les « techniciens » dont ils dénoncent la « froideur inhumaine » et le « technicisme ».

* *

Les critiques adressées à la technique paraissent étranges si l'on convoque l'étymologie: *τέχνη* veut dire « savoir-faire ». Comment pourrait-on être *contre* le savoir-faire, le savoir pratique, l'efficacité ?

Ce n'est pas en fait le savoir-faire que visent les adversaires de la technique, mais le langage conceptuel, la modélisation qui rend compte du monde de telle sorte que l'on puisse agir sur lui ; ils visent la déperdition symbolique, la perte des qualités allusives du langage dont on doit payer l'efficacité dans l'action ; ils visent aussi les attitudes « froides », « inhumaines » de ceux qui se vouent au langage conceptuel. Ils voudraient que l'on pût être pratiquement efficace tout en conservant dans l'action la richesse des connotations, l'ambiguïté suggestive de la langue : mais cela, c'est impossible.

Ces disputes entre « scientifiques » et « littéraires » trahissent une méconnaissance de la respiration de la pensée. Celle-ci a besoin tantôt d'élargir la sphère de ses représentations, et pour cela de laisser aller les associations d'idées qui lui procureront son terrain ; tantôt de construire, sur la base ainsi élaborée, des concepts et structures hypothético-déductives : dans ce second temps elle doit se fermer aux sirènes de l'allusion, éliminer les connotations. Ne vouloir admettre que l'une ou l'autre des deux phases de la démarche,

c'est comme si l'on disait que dans la respiration seule l'inspiration serait légitime, l'expiration étant à proscrire (ou l'inverse). Celui qui applique une telle règle sera bientôt étouffé.

Le flux qui renouvelle et alimente notre pensée passe par le langage connoté, le langage conceptuel permet de mettre en exploitation le stock des représentations ainsi accumulées. Il n'existe pas de stock sans flux qui l'alimente, le flux se perd s'il n'alimente pas un stock.

1.3.3 « Objet »

Dans un « langage à objets » (on dit aussi « langage orienté objets »), on appelle « objet » un petit programme qui contient :

- le nom propre (ou matricule, ou identifiant) qui désigne sans ambiguïté un « individu » du domaine considéré (un client, un produit, un établissement, une machine, une pièce détachée etc.) ;

- diverses variables observées sur cet individu et dont il a été jugé utile de conserver la valeur en mémoire (par exemple date et lieu de naissance, adresse et numéro de téléphone d'une personne ; adresse, activité principale, taille d'un établissement etc.) : on appelle ces variables « attributs » ;

- diverses fonctions qui, appliquées aux « attributs », lancent des traitements produisant d'autres attributs ou encore des messages d'erreur ou d'anomalie (calculer l'âge d'une personne à partir de sa date de naissance et de la date du jour ; mettre à jour la valeur d'un attribut à partir d'une nouvelle saisie ; s'assurer que la saisie est réalisée dans un format conforme, que la donnée a une valeur acceptable etc.) : on appelle ces fonctions « méthodes » et elles transcrivent des « règles de gestion ».

Ainsi l'objet (informatique) représente un être du monde de la nature ; il garde trace de certains de ses attributs (mais non de tous, car tout être naturel possède une infinité d'attributs) ; il leur associe des traitements spécifiques.

Chaque objet est un cas particulier au sein d'une « classe » : par exemple l'objet qui représente *un* client est un cas particulier au sein de la classe « client ». Lorsqu'on définit une classe, on définit la liste des attributs et méthodes que l'on veut connaître sur chacun des individus qu'elle comporte. Lorsqu'on indique les valeurs prises par l'identifiant et les attributs pour un individu particulier on dit que l'on « instancie » la classe, dont l'objet particulier est une « instance ».

Ce jargon s'éclaire si l'on pense à ce qui se passe lorsqu'on fait une enquête statistique (Volle [211]). L'individu appartenant au champ de l'enquête, c'est l'être qu'il s'agit de représenter. Le dessin du questionnaire, c'est la définition de la classe. Remplir le questionnaire, c'est l'« instancier » pour représenter un individu particulier. Les règles de codage et de vérification automatique utilisées lors de la saisie sont des « méthodes » au sens des langages à objets.

Mais « objet » est un faux ami : lorsque l'informaticien l'utilise pour désigner la *représentation* d'un existant, il s'écarte de l'usage courant comme de l'usage philosophique où « objet » désigne un *existant* repéré par la

perception ou visé par l'intention d'un sujet (comme votre voiture, votre montre etc.). Cela conduit l'informaticien à prononcer cette phrase qui fait se hérissier les cheveux du philosophe : « un objet, c'est une abstraction ». C'est presque une tautologie si l'on sait que le terme « objet », pour l'informaticien, désigne une représentation car toute représentation résulte d'une abstraction ; c'est une absurdité si l'on donne au terme « objet » le sens qu'il a dans le langage courant comme en philosophie.

1.3.4 « Donnée » et « Information »

« Frequently the messages have meaning; that is they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem ». (Shannon [185] p. 31.)

La quantité d'information qu'apporte un texte serait, selon la « théorie de l'information » de Shannon, d'autant plus grande que le texte est plus long et moins redondant. Une suite de lettres tirées au hasard ne comporte aucune redondance ; elle contiendrait donc, si on veut la reproduire exactement, plus d'information (au sens de Shannon) que n'importe quel texte de même longueur. « Wan trus be lifx », tapé au hasard sur mon clavier, contiendrait plus d'information que la phrase de même longueur « le pape est mort ».

Pour un lecteur il n'en est pas de même car la seconde phrase a un sens, alors que la première n'en a aucun. C'est que Shannon pensait non à l'être humain, mais à l'ordinateur et aux réseaux : c'est exactement ce qu'il dit dans la citation ci-dessus. Sa théorie est non pas une « théorie de l'information », mais une « théorie des données » ou mieux une « théorie des télécommunications ». L'expression « théorie de l'information » suscite des contresens qui empêchent de distinguer deux concepts également utiles¹⁷.

Ce qui nous est *donné*, c'est le monde de la nature et l'expérience que nous en faisons ; ce que l'on appelle « donnée » en informatique (comme en statistique), c'est le *résultat* d'une observation faite sur l'un des êtres qui existent dans ce monde, autrement dit la mesure d'une variable sur un individu.

Cette mesure n'est pas donnée, mais *obtenue* à l'issue d'un processus d'abstraction qui comporte plusieurs étapes. Nous avons *décidé* :

- (1) d'observer telle population parmi toutes celles que comporte le monde de la nature,
- (2) d'observer sur les individus qui la composent telle sélection parmi l'infinité des variables que l'on pourrait observer sur eux,

17. Pour une analyse de la théorie de Shannon, voir Escarpit [54]. L'article fondateur de Shannon n'a pas pour titre « théorie de l'information », mais « théorie de la communication » : il s'agit en fait d'ingénierie des réseaux de télécommunication (Shannon [184]).

- (3) de coder chaque variable de telle façon (unité de mesure, format, nomenclature pour les variables qualitatives),
- (4) d'identifier tel individu au sein de cette population,
- (5) d'observer sur cet individu la valeur des variables sélectionnées.

Certaines personnes oublient cependant que les « données » résultent d'une observation sélective et les prennent pour la reproduction fidèle et complète de la réalité. D'autres estiment par contre que, du fait de cette sélection, les données seraient irrémédiablement « subjectives » et donc douteuses : pourtant les valeurs qu'elles prennent ne dépendent pas du bon vouloir de l'observateur... Il y a là un nœud de confusions à démêler.

L'informatique enregistre, traite et stocke des données ; l'utilisateur les saisit ou les consulte et lance des traitements qui produisent d'autres données. Les flux qui en résultent dans les processeurs et les réseaux, les stocks qui s'accumulent dans les mémoires, les délais de mise à disposition, le synchronisme des répliquions, la concurrence entre des mises à jour simultanées, tout cela pose des problèmes de *physique*.

La « physique des données » est le domaine propre de la technique informatique. Elle détermine le dimensionnement des ressources (débit des réseaux, puissance des processeurs, taille et délai d'accès des mémoires) qu'elle organise selon l'ordre des performances et des coûts : les mémoires d'accès rapide, coûteuses, seront de faible volume et réservées aux utilisations urgentes, le gros des données étant stocké sur des mémoires moins coûteuses d'accès plus lent. Le système d'exploitation transfère automatiquement les données entre les divers types de mémoire. Le réseau est dimensionné par arbitrage entre le coût du haut débit et le besoin de transferts volumineux et rapides etc.

Les données sont organisées selon des architectures diverses : ici l'on a réalisé un système par objets, là une base de données relationnelles ; ici on a utilisé telle nomenclature ou tel typage, là une autre nomenclature, un autre typage. La communication entre les diverses parties du système d'information demande alors des transcodages et restructurations qui seront effectués par des *interfaces*. Savoir définir ces architectures, savoir interpréter les offres des fournisseurs de solutions, c'est là un métier de spécialiste.

Les données sont parfois fausses : des erreurs se produisent lors de la saisie, ne serait-ce qu'en raison des fautes de frappe. C'est pourquoi les doubles saisies, les recopiations de données à la main, sont un des points de fragilité des systèmes d'information. Par ailleurs lorsqu'une nomenclature change il est parfois nécessaire de procéder à la correction rétrospective des données, et le plus souvent cette correction se paie par des inexactitudes.

* *

Prenons le mot « information » non au sens qu'il a dans la théorie de Shannon, ni à celui qu'il a dans le langage courant (les « informations de 20 heures »), mais au sens étymologique : une information, c'est quelque chose qui vous *in-forme*, qui modifie ou complète la *forme* intérieure de votre

représentation du monde, qui vous *forme* vous-même¹⁸. L'information ainsi conçue a une *signification* : elle suscite une action de la part de celui qui la reçoit, ou du moins elle modifie (*trans-forme*) les conditions de son action future.

Pour pouvoir recevoir de l'information, il faut avoir été formé, et c'est en recevant de l'information que l'on se forme. Certes il faut une amorce pour ce cycle mais elle est enfouie dans les origines de la personne, tout comme l'amorce du cycle de la poule et de l'œuf est enfouie dans les origines de la vie.

Une donnée ne peut donner naissance à une information que si elle est communiquée au destinataire dans des conditions telles qu'il puisse *l'interpréter*, la situer dans son propre monde intérieur et lui attribuer un sens. C'est là le but implicite de nos bases de données, de nos systèmes d'aide à la décision.

Le statisticien connaît le travail qu'il faut fournir pour interpréter les données : les confronter avec d'autres, évaluer des corrélations, revenir sur les définitions et conditions de l'observation etc. Il se publie beaucoup de tableaux de nombres mais peu de personnes disposent de l'arsenal technique nécessaire pour les interpréter. En fait, il faut le dire, personne ne les regarde sauf s'ils sont accompagnés de la synthèse en langage naturel qui permet de les faire parler¹⁹.

Mais les données ne sont pas utilisées principalement pour produire des statistiques : elles servent surtout à traiter des cas particuliers. Des décisions concernant chacun de nous sont prises à partir de dossiers où nous sommes représentés par quelques données plus ou moins bien choisies, plus ou moins exactes. La personne qui contrôlera les traitements automatisés et traitera notre cas pourra-t-elle transformer ces données en informations pour comprendre notre situation ? Ou bien se comportera-t-elle en assistant de l'automate ?

* *

Le mot « comportement » fait passer de la physique des données à la *physique de l'information*. L'utilisateur d'une information *se comporte*. Il n'est pas une chose qui obéirait aux lois de la physique comme le fait le paquet d'octets qui transite par un réseau, car il agira en fonction de ce qu'il a compris.

La physique de l'information ressemble à celle de la circulation routière où les conducteurs, eux aussi, se « comportent » :

18. « Informer » et « instruire » sont très proches : informer, c'est donner une forme ; instruire, c'est donner une structure.

19. On a pu définir ainsi la fonction de la synthèse statistique : « consentir une perte en information (au sens de Shannon) pour obtenir un gain en signification (information au sens étymologique) ». Le travail du statisticien est semblable à celui du typographe : la typographie fait perdre l'information que contiennent l'écriture manuscrite, les corrections etc., mais la mise en forme du texte imprimé facilite sa lecture et aide au dégagement du sens par le lecteur. De même la synthèse statistique attire l'attention sur les données significatives, ce qui invite à négliger celles qui ne le sont pas et prépare les voies de l'interprétation (Volle [212] p. 51).

1) La route qui relie telle banlieue au centre ville est encombrée, on décide de l'élargir : elle sera tout aussi encombrée car comme elle est plus large davantage de personnes prendront leur voiture. De même, vous dimensionnez largement le réseau pour faciliter la tâche des utilisateurs, le nouveau réseau sera tout aussi encombré car de nouveaux usages s'y installeront.

2) Vous affichez sur un panneau de l'autoroute « bouchon à 6 km », certains conducteurs prendront la prochaine sortie et viendront encombrer le réseau des routes secondaires, d'autres resteront sur l'autoroute : ce choix est aléatoire. De même, vous avez installé dans votre entreprise un serveur protégé par un pare-feu modeste : des pirates sauteront cette barrière pour utiliser ce serveur gratuitement. Vous augmentez la puissance du pare-feu : vos utilisateurs seront gênés etc.

La conception d'un système d'information doit anticiper le comportement des utilisateurs tout comme ceux qui conçoivent un réseau routier anticipent le comportement des conducteurs. Le système d'information influence en retour les comportements. Vous avez organisé de telle façon l'annuaire de l'organisation, découpé de telle sorte les zones géographiques, choisi telle nomenclature de produits : c'est ainsi que votre entreprise parlera, s'organisera, communiquera. Certaines décisions, prises à la va-vite par un groupe de travail, détermineront à long terme le cadre des représentations selon lesquelles l'entreprise définira ses priorités. La délimitation des populations décrites dans le système d'information, la gestion des identifiants, des nomenclatures, des classes d'objets, bref l'ensemble des opérations que l'on a coutume de nommer « administration des données » ou « gestion du référentiel », conditionnent la physique de l'information.

1.3.5 « Numérique »

Que d'émotions autour du mot « numérique » ! La « fracture numérique » mettrait en péril la cohésion sociale ; l'ordinateur constituerait un danger majeur pour les arts, qu'il priverait de leur âme en les « numérisant ». Un frisson parcourt l'échine du « littéraire » confronté à la « froideur » des mathématiques et de l'abstraction.

Chacun est libre de ses goûts et dégoûts, mais on ne peut admettre que s'installe une erreur de jugement fondée sur un pur effet de vocabulaire. Nous devons donc développer ici l'analyse d'une erreur triviale : des esprits par ailleurs distingués la commettent. Comme ils sont éloquentes et écoutés, il en résulte dans les idées un désordre dévastateur²⁰.

La confusion est pire encore lorsqu'on utilise le terme anglais « digital ». « Digit » signifie « chiffre » en anglais, mais « digital » désigne en français ce qui est relatif au doigt. L'expression « son digital » ahurit le badaud - c'est sans doute le but visé par des vendeurs péremptoires, fût-ce au prix d'une détérioration de la langue.

20. Virilio [209] annonce « la fin du langage ». Heureusement cette prophétie effrayante n'est étayée par aucun argument, pas plus que la dénonciation du « cybermonde » que Virilio fonde sur la ressemblance entre les mots « interactivité » et « radioactivité ». Certains voient de la profondeur dans ces calembours.

À propos de la « fracture numérique »

Cette expression, aussi disgracieuse que « son digital », désigne la différence sociale qui s'instaurerait entre ceux qui maîtrisent l'ordinateur et ceux qui, ne le maîtrisant pas, risqueraient l'exclusion.

Certains de ceux qui manifestent ainsi leur souci envers les exclus potentiels se flattent de n'avoir aucune pratique de l'ordinateur. Sans doute se considèrent-ils comme des privilégiés au grand cœur préoccupés par l'inégalité qui les sépare de leurs inférieurs, inégalité qu'ils savourent tout en la déplorant. Comme il est délicieux de gagner à la fois sur le tableau social et sur le tableau moral !

Cela rappelle les « préceptes orgueilleusement humbles d'un snobisme évangélique » que la princesse de Parme inculque à sa fille dans *À la recherche du temps perdu*^a.

Pourtant personne, quel que soit son niveau social, n'est embarrassé pour utiliser un distributeur automatique de billets ou un téléphone portable, outils *high tech* s'il en est. Les assistantes sont plus expertes que leur patron dans l'utilisation de l'ordinateur. Les personnes les plus calées en informatique, celles qui maîtrisent les langages de programmation et les architectures (savoir dont l'acquisition demande des années de formation), sont des cadres moyens à qui leur spécialité procure la légitimité qui leur est refusée par ailleurs. Les cadres supérieurs, sauf exception, ne feront pas l'effort de se qualifier en informatique tant que leur légitimité ne sera pas écornée par leur incompetence, et cette heure-là n'a pas encore sonné en France.

La vraie « fracture numérique » se trouve entre les dirigeants et l'informatique, et non tout près de la frontière de l'exclusion sociale. Bien sûr personne ne peut utiliser un ordinateur sans un minimum d'explications et de pratique ; mais il ne faut pas prétendre que certains souffriraient, à cet égard, d'un handicap social : il est beaucoup plus difficile d'apprendre à parler en bon français que d'apprendre à utiliser l'ordinateur (l'apprentissage de la programmation, lui, est par contre aussi difficile que celui d'une langue naturelle).

^a « Rappelle-toi que si Dieu t'a fait naître sur les marches d'un trône, tu ne dois pas en profiter pour mépriser ceux à qui la divine Providence a voulu (qu'elle en soit louée !) que tu fusses supérieure par la naissance et par les richesses. (...) Sois secourable aux malheureux. Fournis à tous ceux que la bonté céleste t'a fait la grâce de placer au-dessous de toi ce que tu peux leur donner sans déchoir de ton rang, c'est-à-dire des secours en argent, même des soins d'infirmière, mais bien entendu jamais d'invitation à tes soirées, ce qui ne leur ferait aucun bien, mais, en diminuant ton prestige, ôterait de son efficacité à ton action bienfaisante. » (Marcel Proust, *À la recherche du temps perdu*, *Le côté de Guermantes* II, 1921 ; Robert Laffont, collection « Bouquins » 1987, vol. 2 p. 352).

Il est vrai que dans l'ordinateur, au cœur du processeur qui effectue les opérations, n'existent que des oscillations rapides entre deux niveaux de tension électrique qui servent à coder des 0 et des 1, des « bits ». L'information que l'utilisateur traite (texte, images, calculs, sons) est transcrite par une cascade de codages qui la traduisent ou l'interprètent pour parvenir au microcode, écrit en bits, que le processeur pourra exécuter.

Ce codage a-t-il une influence sur l'information? Non, car il ne fait que la transcrire. Si je tape une fable de La Fontaine sur mon clavier, son texte s'affichera sur l'écran avec toute sa puissance évocatrice. Que les caractères soient codés en octets (huit bits) n'enlève rien à son contenu et les conventions de traitement de texte que j'utilise pour le mettre en page, elles aussi transcrites en bits, ne font que faciliter sa lecture.

L'ordinateur, faisant ici fonction de machine de traitement de texte, est incapable d'interpréter ce texte mais il aide à sa présentation. On ne peut pas dire que la fable de La Fontaine soit « numérisée » : elle reste un texte poétique qui vise, par ses suggestions et sa musicalité, à émouvoir le lecteur et éveiller son intelligence.

Parodions, en le transposant, le raisonnement des critiques du « numérique »²¹ : « Comment voulez-vous que l'écriture puisse reproduire la richesse et les nuances du langage humain? Comment pourrait-on décrire des couleurs quand on écrit en noir sur du papier blanc? » etc. Cette transposition met à nu le procédé qu'utilisent des sophistes pour susciter doute et perplexité : on feint de croire, en confondant les diverses couches du processus, que la physique du support rétroagit sur le contenu du texte. Quand Marshall McLuhan dit *the medium is the message* (McLuhan [128]) cela ne signifie pas « les ondes électromagnétiques constituent le message de la télévision », mais « les conditions économiques, sociales, de la production des programmes télévisuels ont sur leur contenu une influence qui peut être déterminante », ce qui est tout différent.

C'est en considérant les conditions pratiques, sociales, culturelles, économiques de la mise en œuvre de l'APU, de l'*utilisation* de l'ordinateur, que l'on peut raisonner sur ses apports, leurs limites et leurs dangers. Le mot « numérique » ne donne pas la clé de ce raisonnement : comme un épouvantail, il inhibe le discernement. Des personnes par ailleurs compétentes raisonnent mal quand elles parlent de la numérisation²² : elles ne voient pas que l'architecture en couches de l'ordinateur implique une diffé-

21. Leur voix, quand ils téléphonent, est soumise au codage MIC qui la transforme en un flux de 64 000 bits par seconde. Il en est de même, avec un débit plus élevé, pour la musique lorsqu'ils écoutent un disque compact. Leur parole, la musique, sont-elles pour autant « numériques »?

22. « L'ordinateur est la matérialisation de la logique mathématique : ils ont connu des développements historiques conjoints. Aux fondements de ces développements se trouve le principe d'identité. L'ordinateur calcule 0 et 1 mais ne sait faire que cela. Tout, en effet, est ramené à des 0 et des 1 afin que le courant électrique passe ou ne passe pas. L'ordinateur oblige à faire des modèles entièrement logiques. Il fonctionne comme un principe de réalité technico-logique, garant de la cohérence des modèles : un producteur de modèles hyperrationnels. » (Francis Pavé, « Transformation des représentations et résistance aux changements », conférence à l'école d'été 1998 de l'IUFM de Franche-Comté). Mais quel est le modèle « hyperrationnel » à l'œuvre quand on utilise un traitement de texte?

rence de nature entre ce que fait l'utilisateur et ce que fait la machine. Que penseraient-elles si on disait de leurs écrits « ce sont des signes noirs dessinés sur du papier blanc », ou de leurs paroles « ce sont des ondes sonores que propagent les variations de la pression de l'air » ?

L'ordinateur est pour nos sociétés une innovation aussi importante que le furent, à d'autres époques, l'invention de l'écriture ou de l'imprimerie : il modifie les conditions de création, classement, recherche et traitement des données et documents. Il ne sera pas facile d'apprendre à s'en servir, à éviter ses effets pervers. Mais avons-nous vraiment maîtrisé l'écriture, l'impression ? Savons-nous vraiment lire et écrire (et compter) au sens non de la performance, mais de la vie intellectuelle et de la vie sociale ? faut-il donc que l'ordinateur nous inspire plus de craintes que la presse, les médias ou même, tout simplement, la parole ?

1.4 Modèle en couches

Le « modèle en couches » a été conçu par des techniciens pour pouvoir penser un objet du monde de la nature dont le fonctionnement articule plusieurs logiques. Ainsi dans un ordinateur se produisent des phénomènes électromagnétiques (des électrons se déplacent, des tensions se modifient) ; ces phénomènes sont traduits en 0 ou 1 ; les suites binaires représentent des instructions ou des données ; les règles qui gouvernent le codage des données et instructions sont fournies par les langages de programmation (Sethi [182]) ; le langage le plus proche de la machine étant incommode pour l'être humain, des « langages de haut niveau » sont élaborés pour faciliter la programmation ; enfin, des « applications » sont programmées pour satisfaire les utilisateurs. Le mouvement des électrons ne peut pas être décrit selon la même grille conceptuelle que les applications, ni que le « compilateur » qui traduit un langage de haut niveau en instructions exécutables, mais tous ces phénomènes jouent conjointement (Tanenbaum [199]).

Le modèle OSI de l'ISO²³ représente en sept couches le fonctionnement d'un réseau de télécommunications, de la couche physique (transport des bits) jusqu'à la couche « application » (Tanenbaum [200]).

Le modèle en couches ne s'applique pas qu'à la seule technique : des philosophes l'ont utilisé. Saussure ([179] p. 27) a décrit les couches psychologique, linguistique, neurologique, phonétique, sonore etc. par lesquelles passe la conversation entre deux personnes. On peut citer aussi Karl Popper et son « modèle des trois mondes » qui distingue et articule le « monde 1 » des objets et forces physiques, le « monde 2 » des états mentaux et le « monde 3 » des symboles, théories, problèmes, langages etc. (Popper [163] ; traduction : *La connaissance objective*, Flammarion 1998, p. 181). Maurice Blondel évoque un modèle en couches lorsqu'il dit, dans une phrase d'une remarquable densité, « entre (la science, la morale et la métaphysique), il

23. « Open Systems Interconnection » publié de 1977 à 1986 par l'International Standard Organization.

n'y a point de contradiction, parce que là où l'on a vu des réalités incompatibles il n'y a (...) que des phénomènes hétérogènes et solidaires » (Blondel [16] p. XXII).

* *

Michel Foucault a décrit l'événement qui, à la fin du XVIII^e siècle, marqua le passage de la représentation classificatrice à une représentation *organique* de la nature : « Cuvier fera main basse sur les bocaux du Museum, il les cassera et disséquera toute la grande conserve classique de la visibilité animale (Foucault [61] p. 150.) ».

La classification des êtres vivants est une opération ensembliste : l'évaluation des ressemblances et différences permet de définir une « distance » puis une relation de ressemblance entre espèces, les espèces semblables étant celles qui sont proches selon la distance choisie (Lecointre et Le Guyader [78]). L'approche organique considère par contre non des collections d'êtres semblables, mais l'articulation des divers organes qui remplissent des fonctions complémentaires dans un seul et même organisme.

Si l'on peut considérer un être vivant comme un ensemble de cellules, il sera souvent plus pertinent de le représenter comme un organisme : l'appartenance d'un bras ou d'un poumon au corps d'un animal, l'appartenance de la direction commerciale ou de la direction des ressources humaines à une entreprise, ne sont pas de type ensembliste (ce ne sont pas des inclusions), mais de type organique (l'organe a une fonction spécifique, articulée avec les fonctions des autres organes). De même on peut voir un système d'information comme ensemble de lignes de code source, mais il sera souvent plus pertinent de le représenter comme un organisme.

Le modèle en couches, qui articule des logiques différentes, relève de l'approche organique. Bien qu'il soit d'origine technique, il constitue une innovation philosophique. Il ne semble pas cependant que les philosophes aient vu dans ce modèle un outil au service du travail intellectuel, même si quelques-uns d'entre eux l'utilisent à l'occasion sans paraître le savoir.

Ce modèle est à l'œuvre dans la pensée quotidienne. Dans un paysage comme celui de *La montagne Sainte-Victoire vue de Bellevue*, de Cézanne, on voit une montagne (géologie) ; des arbres (botanique) ; un pont, des routes, des champs cultivés (action humaine) ; des nuages (météorologie). Ce paysage articule ainsi des êtres qui relèvent chacun d'une logique et d'un rythme spécifiques. L'unité de temps est pour la géologie le million d'années, pour la couverture végétale le millénaire ou le siècle, pour les constructions humaines le siècle ou la dizaine d'années, pour le déplacement des animaux ou des nuages l'heure ou la journée. Il est impossible de rendre compte d'un tel paysage selon une seule logique, si ce n'est celle de la superposition des logiques et des rythmes qu'il articule.

1.4.1 Couches de l'ordinateur

Le processeur d'un ordinateur fonctionne très vite mais ne peut reconnaître et exécuter qu'un nombre limité d'instructions élémentaires comme

« copier dans ce registre l'enregistrement stocké dans la mémoire à telle adresse », « additionner le contenu de ces deux registres et noter le résultat dans ce troisième », « dire si le contenu de ce registre est égal ou non à zéro », « dire si le contenu de ce registre est supérieur à celui de cet autre registre », « stocker le contenu de ce registre dans un enregistrement ayant telle adresse » etc.

L'ensemble de ces instructions constitue le *langage machine* que nous noterons L_1 . L_1 est pauvre : ses instructions s'écrivent sous forme de 0 et de 1 et il serait très pénible de programmer dans un tel langage.

On écrit donc en L_1 un ensemble d'instructions plus riche et plus proche du langage naturel et on construit à partir de ces instructions un nouveau langage L_2 (« assembleur²⁴ ») plus facile à utiliser.

Il existe deux façons d'exécuter un programme écrit en L_2 :

- un programme écrit en L_1 (nommé traducteur ou compilateur²⁵, ces deux termes sont synonymes) remplace la suite des instructions du programme en L_2 par une suite équivalente d'instructions en L_1 (Aho, Sethi et Ullman [4]). L'ordinateur exécutera ensuite le programme en L_1 ainsi obtenu.

- un programme en L_1 (interpréteur), après avoir lu une instruction en L_2 , exécute immédiatement la séquence en L_1 équivalente. L'interprétation évite de générer un programme en L_1 mais son exécution est plus lente que celle du programme compilé en L_1 .

Du point de vue de l'utilisateur, l'ordinateur doté d'un compilateur ou d'un interpréteur obéit aux instructions écrites en L_2 aussi docilement que si elles étaient écrites en L_1 : l'ensemble constitué de la machine physique M_1 , des langages L_1 et L_2 et de l'interpréteur (ou du compilateur) de L_2 en L_1 constitue la « machine virtuelle » M_2 . Pour celui qui écrit un programme en L_2 , M_2 est aussi « réelle » que M_1 et son usage est plus commode.

Il existe cependant des limites à la complexité acceptable pour un traducteur ou un interpréteur. L_2 , bien que plus commode que L_1 , peut ne pas être encore le langage le plus convenable pour programmer. On écrit donc en L_2 un ensemble d'instructions et on construit un nouveau langage plus commode, L_3 , qui définit la nouvelle machine virtuelle M_3 .

24. Pour expliciter la façon dont la machine exécute les algorithmes et donner à ceux-ci la forme précise et complète que seule procure l'écriture d'un programme, Donald Knuth a défini l'ordinateur MIX (1009 en chiffres romains) et le langage assembleur MIXAL (Knuth [105] p. 124 et 144). Le lecteur peut ainsi, moyennant un effort d'attention, se représenter exactement la façon dont l'algorithme est exécuté et en déduire la précision du calcul, sa durée d'exécution etc. Pour tenir compte de l'évolution des processeurs Knuth a aussi défini le MMIX (2009) qu'il a décrit sur son site www-cs-faculty.stanford.edu/~knuth/.

25. Le mot « compilateur » date de 1951, avant la mise au point des langages de programmation proprement dits. Il s'agissait d'appeler dans un programme des procédures préprogrammées (*subroutines*) comme par exemple celles qui servent à calculer un sinus ou un logarithme. Le programme se présentait alors comme une *compilation* de procédures. « Compilateur » est resté et a désigné, par la suite, le logiciel qui traduit le code source en un code exécutable par la machine. « The reason it got called a compiler was that each subroutine was given a "call word," because the subroutines were in a library, and when you pull stuff out of a library you compile things. » (Hopper [85] p. 10).

Langages et machines virtuelles s'empilent ainsi jusqu'à la couche n (figure 1.8).

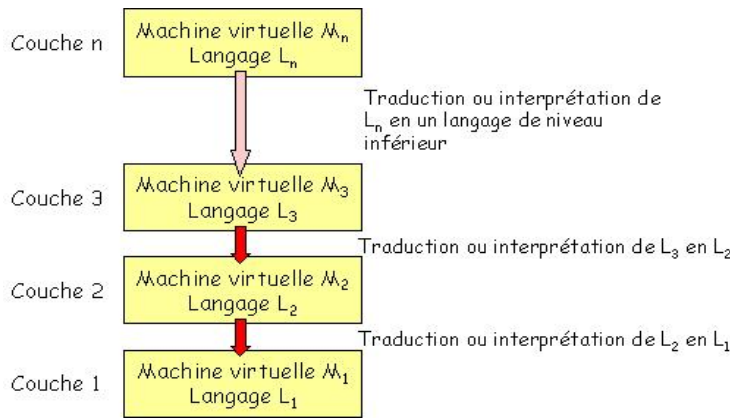


FIG. 1.8 – L'empilage des « machines virtuelles »

Dans les années 1940, les ordinateurs n'avaient que deux couches (ou « niveaux ») : le niveau machine, dans lequel on programmait, et le niveau physique qui exécutait les programmes. Les circuits de ce dernier niveau étaient complexes, difficiles à construire et peu fiables. Maurice Wilkes conçut en 1951 un ordinateur à trois couches afin de pouvoir simplifier le matériel. La machine disposait d'un interpréteur qui exécutait les programmes écrits dans le langage machine. Le matériel ne devait plus alors exécuter que des microprogrammes dont le répertoire d'instructions était limité.

Les assembleurs et compilateurs furent écrits dans les années 1950. En 1954 le premier système d'exploitation (gestion automatique de la mémoire, gestion des répertoires de fichiers) fut conçu par Gene Amdahl pour l'IBM 704. La plupart des ordinateurs actuels possèdent six couches (Tanenbaum [198] ; figure 1.9).

Quelqu'un qui utilise un ordinateur pour faire du traitement de texte met en œuvre un programme dont les instructions parviennent à la couche physique (processeur et mémoires) après une cascade de traductions et d'interprétations. L'utilisateur peut ignorer ces opérations : pour lui, et dans le cadre de cette application, l'ordinateur fonctionne comme une machine de traitement de texte et il n'a à connaître que cette seule machine (c'est pourquoi il est inexact de dire que son activité est « numérisée » même si l'ordinateur, pour exécuter ses ordres, les transcrit sous la forme de suites de 0 et 1).

Une instruction du niveau applicatif, traduite ou interprétée en cascade, engendre dans la couche microprogrammée de nombreuses instructions. Le temps d'exécution d'un programme peut s'évaluer en additionnant les temps unitaires : ce calcul est une étape importante de l'évaluation d'un algorithme (Knuth [105] vol. 1 p. 170).

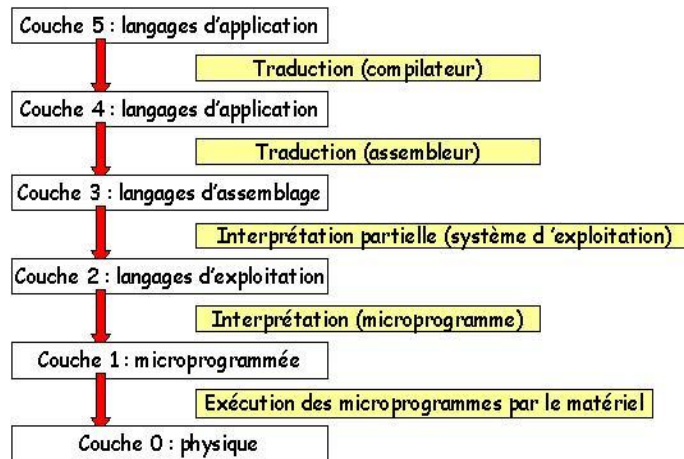


FIG. 1.9 – Couches de l'ordinateur

Si l'on veut optimiser le délai de traitement il faut maîtriser la succession des instructions et pour cela programmer dans les « couches basses ». Cependant l'accroissement des performances des processeurs et de la taille des mémoires a réduit l'utilité de cette optimisation, sauf pour certaines applications où la rapidité est cruciale. Les programmes sont donc presque tous écrits dans des langages de niveau élevé ; l'art du programmeur s'appuie alors sur sa connaissance des propriétés du langage de programmation, qui doit être suffisante pour anticiper les conséquences physiques des choix faits dans les couches hautes.

L'utilisateur d'un ordinateur n'a à se soucier que du service rendu par la couche dont il se sert et qu'il considère comme une ressource physique (« matériel et logiciel sont équivalents »). Celui qui conçoit un ordinateur optimise l'utilisation des ressources couche par couche, et non en considérant le processus d'ensemble (Kidder [101]) : la simplicité que l'on gagne ainsi fait plus que compenser une éventuelle perte en efficacité physique.

1.4.2 Portée du modèle en couches

Pour montrer à quel point le modèle en couches est présent (le plus souvent de façon implicite) dans notre vie quotidienne, considérons la conversation entre deux personnes (Saussure [179] p. 27). Elle emprunte les couches suivantes (figure 1.10) : conception et compréhension des idées à communiquer (logique) ; codage et décodage de ces idées dans un langage (sémantique) ; codage et décodage de ce langage dans des phonèmes (phonétique) ; mécanismes de l'articulation et de l'audition (physiologie) ; émission et réception d'ondes sonores (physique).

Deux personnes qui conversent font fonctionner ces diverses couches alternativement dans les deux sens, mais ne s'intéressent qu'aux idées qu'elles émettent ou reçoivent. Les couches inférieures n'attireront l'attention que

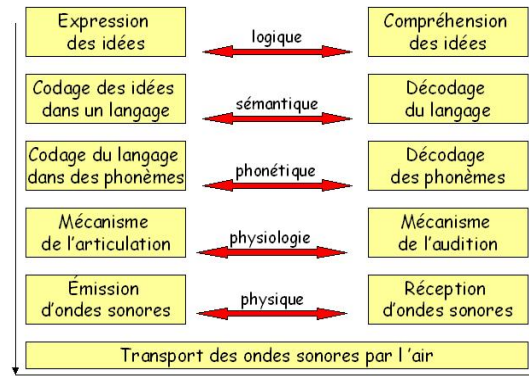


FIG. 1.10 – Couches de la conversation

si elles fonctionnent mal : si l'un des locuteurs parle une langue ignorée de l'autre, ou encore s'il est sourd ou aphasique ; les ondes sonores ne sont utilisables que si les deux interlocuteurs sont assez proches l'un de l'autre et si le milieu sonore ambiant n'est pas trop perturbé par le bruit.

Il suffit d'ajouter quelques couches en bas du modèle pour représenter une conversation téléphonique : l'onde sonore est codée sous la forme d'une onde électromagnétique par le poste téléphonique, numérisée dans le commutateur de départ, acheminée par le réseau, décodée à l'arrivée.

Un modèle en couches décrit un empilage de conditions *toutes également nécessaires*. Or si, dans un produit qui comprend des 0 et des 1, tous les termes sont égaux à 1 sauf un seul qui est égal à 0, le produit est nul ; de même, lorsqu'un phénomène obéit à plusieurs conditions simultanément nécessaires elles doivent toutes être respectées pour que le phénomène puisse se produire. Dès lors il serait vain de se demander laquelle de ces couches est « la plus importante », de tenter d'établir entre elles une hiérarchie.

On cherche parfois à déterminer l'origine et le responsable de la réussite d'un processus : cet effort est vain si le processus a traversé plusieurs couches toutes nécessaires. Si le stratège doit concentrer son attention sur la question la plus importante du moment, il n'en doit pas moins rester attentif à la coopération de diverses couches indispensables au processus ; son aptitude à coordonner des activités diverses se fonde sur la connaissance de leur articulation. Cette coordination n'est pas mélodique (une seule ligne dont il faut suivre le déroulement dans le temps) mais polyphonique (plusieurs lignes qu'il faut suivre ensemble et qui s'articulent entre elles).

1.5 Vers l'ordinateur individuel

Le silicium (Si) est, après l'oxygène, le constituant le plus massif et donc le plus répandu de la Terre. Le silex, constitué de silice pure (dioxyde de silicium SiO_2), a servi voici 2,5 millions d'années de matière première aux outils, armes et parures fabriqués par *Homo habilis* en Afrique de l'Est

La conception d'un ordinateur

Tracy Kidder ([101]) a décrit les étapes de la conception du mini-ordinateur Eclipse MV/8000 lancé par Data General en avril 1980.

Projet, réalisation, intégration, débogage, sont autant d'aventures qui s'articulent lors de la conception sous une contrainte de délai toujours présente, toujours violée, toujours renouvelée. La mise au point des langages et celle du matériel se renvoient la balle ; des spécialités diverses, aux sociologies contrastées, s'entrecroisent. Assurer la compatibilité du produit avec la gamme de l'entreprise, assurer le respect des interfaces, tout cela suppose des compromis fondés sur le sens du marketing et du possible technique.

Les hommes qui font ce travail sont des ingénieurs au sens plein du terme : ils réalisent de façon ingénieuse un travail d'ingénierie. Ils se trouvent au front de taille de l'action, au point où elle pose des problèmes intellectuels d'une extrême subtilité, où elle rencontre la limite de notre connaissance de la nature. Ce sont des rêveurs pratiques, des passionnés. L'argent n'est pas leur moteur, ils le disent avec insistance et s'en étonnent eux-mêmes : ils ressemblent plutôt à des alpinistes qui se lancent des défis.

La limite qu'ils cherchent, ce n'est pas tant une limite individuelle qu'une limite humaine : ils veulent voir jusqu'où un être humain peut aller dans la clarté de la conception, la maîtrise de la complexité, la concentration de la pensée, la rapidité d'exécution, l'élégance de la démarche.

L'animateur de l'équipe, Tom West, manifeste son respect pour ses collaborateurs en étant froid et distant. « Pas de tapes dans le dos », dit-il ; mais il leur délègue des responsabilités qu'ils n'auraient jamais pu avoir ailleurs et conduit en stratège l'affaire vers le succès en assumant le poids des reproches et des frustrations de l'équipe.

On trouve dans ce livre une excellente description intime de l'ordinateur, de la façon dont il est perçu, pensé, voulu par ceux qui le conçoivent. Les pages consacrées au microcode, à l'assembleur, à la conception des cartes, sont très pédagogiques. L'articulation des couches de l'ordinateur devient ainsi intuitive, palpable.

Les concepteurs de l'ordinateur en restent cependant à « la machine ». Ils ne s'intéressent pas à la diversité de ses utilisations potentielles (c'est une autre question, qui devra être traitée par d'autres personnes à un autre moment), mais ils veillent à la doter des performances qui lui permettront d'être efficace dans la plus grande diversité d'utilisations possible.

Ils créent et balisent ainsi l'espace où devra se conduire une recherche postérieure à la leur : la recherche sur les usages, sur l'assistance qu'apporte « la machine » au travail humain. Le livre, qui s'arrête au moment où « la machine » quitte les mains des ingénieurs pour passer dans celles des vendeurs (la rencontre entre ces deux populations est décrite avec humour), ne considère pas cette deuxième recherche, qui se déroulera chez les utilisateurs, les clients, les intégrateurs.

(Gille [66]). Par la suite l'être humain a abondamment utilisé le silicium sous forme de céramique, de brique, puis de verre et de porcelaine.

Le silicium pur cristallisé, produit par l'industrie chimique à partir de verres de silice (quartz) sert aujourd'hui de matière première aux « puces » de la microélectronique. Le silicium est ainsi intimement lié à l'histoire de notre espèce. Il est peu coûteux en raison de son abondance ; ses propriétés électroniques expliquent à la fois le tranchant des arêtes du silex et les performances des circuits intégrés.

Du processeur au microprocesseur

Le processeur est la partie centrale et aussi la partie la plus complexe d'un ordinateur. Il comprend les registres et la machine qui exécute les opérations commandées par le programme ^a. Les ingénieurs qui conçoivent un processeur doivent faire des choix en tenant compte d'un grand nombre de facteurs économiques et techniques : il en est résulté une diversité de solutions qui a rendu les premiers ordinateurs mutuellement incompatibles. Pour un utilisateur, il était coûteux de changer de fournisseur car il aurait fallu réécrire les programmes afin de les adapter à une autre machine.

Le microprocesseur, processeur intégré sur une seule puce, a introduit un standard de fait : Intel vend à qui veut les acheter ses microprocesseurs qui deviennent alors de simples composants et dispensent le client des études les plus complexes et les plus coûteuses (Brulé [28] p. 282).

^aChargement dans un registre d'une valeur contenue dans une mémoire, stockage en mémoire de la valeur contenue dans un registre, opérations arithmétiques sur les registres, calculs sur les adresses, comparaison entre une valeur en mémoire et une valeur dans un registre, sauts d'instructions etc.

L'exploitation des potentialités du silicium a requis la maîtrise de plusieurs techniques délicates. Elle a donc été progressive. Cette progressivité explique la « loi de Moore » (doublement des performances tous les 18 mois) qui elle-même explique pour partie la montée des performances et la baisse du prix des ordinateurs, puis l'émergence de l'ordinateur individuel, enfin la pénétration et la généralisation de l'ordinateur dans les entreprises et la société.

* *

En 1947, trois chercheurs des Bell Labs d'AT&T (John Bardeen, Walter Brattain et Robert Shockley) découvrent le « transistor » (*transfer resistor*). Leur invention s'appuie sur le fait qu'il est possible de contrôler le flux d'électricité dans le germanium (remplacé ensuite par le silicium) en faisant en sorte que certaines zones soient conductrices et d'autres isolantes, d'où le terme « semi-conducteur ». Comparé au tube à vide, technique jusqu'alors dominante, le transistor se révélera plus fiable, moins consommateur d'énergie et susceptible de miniaturisation.

Les premiers transistors étaient faits à la main dans des conditions rustiques si on les compare aux « salles blanches » actuelles. Les rendements étaient bas (de 20 à 30 %) et les performances très variables. Les progrès techniques ont donc d'abord concerné la maîtrise du processus de production.

Pendant les années 1950 une nouvelle industrie se crée. La fabrication des semi-conducteurs fait d'importants progrès : la *diffusion* consiste à diffuser des impuretés (« dopants ») sur leur surface, ce qui supprime le processus fastidieux d'ajout de diverses couches de matériaux isolants et conducteurs sur le substrat. Des *techniques photographiques* permettent de projeter sur le semi-conducteur le dessin de masques compliqués de telle sorte que la diffusion ne se produise que sur les surfaces souhaitées.

Le circuit intégré (circuit électronique comportant plusieurs transistors sur une même pièce de matériau semi-conducteur) est inventé en 1958 par Jack Kilby²⁶. Jean Hoerni, de Fairchild Semiconductor, introduisit en 1959 une autre innovation essentielle en observant que les techniques de diffusion et de photographie permettaient de se libérer des complications du transistor à trois dimensions et de dessiner des *transistors plans* (« planars »). Il devenait alors possible de faire les connections électriques non plus à la main, mais en déposant par condensation de vapeur métallique un film sur les parties appropriées du semi-conducteur. Moore a fait de l'invention du planar le point de départ de la « loi de Moore ». Fairchild Semiconductor produisit le premier circuit intégré utilisant cette technique en 1961.

Ces techniques permettront de passer de la production manuelle à la production industrielle de série avec une meilleure qualité. Les progrès des processus de production et des techniques se sont ensuite poursuivis. Les méthodes photographiques sont devenues de plus en plus précises grâce notamment à l'emprunt de techniques de photolithographie conçues pour l'imprimerie.

Les premiers circuits intégrés étaient des mémoires. Le microprocesseur, qui est avec l'ampoule électrique et le téléphone considéré comme l'un des plus grands apports de la technique américaine, sera inventé en 1971 par Intel.

1.5.1 Origines et évolution du microprocesseur

Intel a été créée en 1968 à Santa Clara, dans la Silicon Valley, par Robert Noyce, Gordon Moore et Andy Grove. Ces trois ingénieurs venaient de quitter Fairchild Semiconductor à la suite d'un désaccord sur la stratégie.

Sur la base d'un business plan d'une page tapé à la machine par Robert Noyce (voir page 52), Art Rock, *venture capitalist* de San Francisco qui avait financé le démarrage de Fairchild et connaissait les qualités professionnelles de ces trois ingénieurs, avança les 2,5 millions de dollars nécessaires²⁷.

26. Le premier circuit intégré, présenté le 12 septembre 1958, comportait deux transistors montés sur une barre de germanium.

27. Cet épisode est caractéristique des relations entre capital risque et ingénieurs dans la

Intel a d'abord produit des mémoires. Busicom, fabricant japonais de machines à calculer, lui demande en 1969 de mettre au point douze circuits intégrés pour assurer les fonctions de ses machines. Un ingénieur d'Intel, Ted Hoff, proposa de concevoir un circuit intégré programmable unique pour réaliser l'ensemble de ces fonctions. Noyce et Grove approuvèrent cette solution, ainsi que Busicom. Le travail fut réalisé en neuf mois sous la direction de Federico Faggin.

Le 4004 est annoncé en novembre 1971. *C'est le tout premier microprocesseur*. Il comprend 2300 transistors, tourne à 108 kHz et a autant de puissance de calcul que l'ENIAC. Il utilise un chemin de données à 4 bits (quatre chiffres binaires sont traités en parallèle), ce qui le rend utilisable pour des automatismes ou des caulettes mais non pour construire un ordinateur.

Intel rachète pour 60 000 \$ les droits de Busicom sur le 4004 (peu après, Busicom fera faillite). Il faut trouver d'autres clients. Intel se lance dans une communication active afin de convaincre la profession d'écrire des logiciels pour le 4004 et de l'utiliser pour des automatismes comme ceux qui assurent le contrôle des feux de circulation.

En avril 1972 Intel produit le 8008. C'est un processeur à 8 bits, ce qui permet à quelques pionniers de construire un ordinateur autour du microprocesseur.

En 1973 sort le Micral de R2E; la revue *Byte* le baptise en juin 1973 du terme « microcomputer ». *Le micro-ordinateur est né*, mais il ne rencontrera le succès commercial qu'avec l'Altair (1974) qui utilisera le 8080 d'Intel sorti en 1974, et surtout avec l'Apple II (1977) qui utilisera le 6502 de MOS Technologies, sorti en 1976.

En 1974 Motorola, entreprise d'électronique créée en 1928 à Chicago et qui s'est spécialisée dans les circuits intégrés, produit son premier microprocesseur, le MC6800 à 8 bits. Il comporte 4 000 transistors. Apple adoptera le processeur 16 bit 68000 de Motorola, sorti en 1979, pour le Lisa (1983) puis pour le Macintosh (1984).

* *

Intel et Motorola produisent des microprocesseurs mais ils ne sont pas les seuls : on peut citer AMD (Advanced Micro Devices) qui produit depuis 1991 des clones de microprocesseurs Intel, ainsi qu'IBM qui produit les PowerPC. Certaines entreprises dites « fabless » (sans fabrication) conçoivent des microprocesseurs qu'elles font fabriquer sur plan par un « fondeur » : c'est le cas de Sun qui fait fabriquer le SPARC par Texas Instruments, de Silicon Graphics avec le MIPS etc.

Le Pentium 4 de 2000 possédait 42 millions de transistors et tournait à 1,5 GHz. Il permettait à l'utilisateur de produire des films, diffuser de la

Silicon Valley. L'investisseur n'est pas seulement un financier : c'est un *entrepreneur*, un expert qui connaît les personnes et les techniques, raisonne sur le marché des produits et anticipe leur évolution (Aoki [5]). Pour un contraste avec le cas de la France, on peut lire les aventures de Guillaume Portes (un « Bill Gates français ») imaginées par Jean-Pierre Brulé ([28] p. 307).

télévision sur l'Internet, communiquer par la voix et l'image, afficher des graphiques 3D en temps réel, coder de la musique au format MP3, et faire tourner simultanément plusieurs applications multimédia tout en restant connecté à l'Internet²⁸.

1.5.2 La loi de Moore

En 1965 Gordon E. Moore était directeur de la recherche et du développement à Fairchild Semiconductor. Préparant un exposé sur l'évolution des performances des mémoires, il constata que la capacité des « puces » avait à peu près doublé chaque année de 1959 à 1965. Il fit l'hypothèse que cette tendance allait se poursuivre. C'est cette hypothèse que l'on appelle « loi de Moore » ; elle était hardie, puisque Moore ne disposait que de cinq observations pour l'étayer.

Le Business Plan d'Intel, 1968

The company will engage in research, development, and manufacture and sales of integrated electronic structures in order to fulfill the needs of electronic systems manufacturers. This will include thin films, thick films, semiconductor devices, and other solid states components used in hybrid and monolithic integrated structures.

A variety of processes will be established, both at a laboratory and production level. These include crystal growth, slicing, lapping, polishing, solid state diffusion, photolithographic masking and etching, vacuum evaporation, film deposition, assembly, packaging, and testing, as well as the development and manufacture of special processing and testing equipment required to carry out these processes.

Products may include diodes, transistors, field effect devices, photo sensitive devices, photo emitting devices, integrated circuits, and subsystems commonly referred to by the phrase "large scale integration". Principle customers for these products are expected to be the manufacturers of advanced electronic systems for communications, radar, control and data processing. It is anticipated that many of these customers will be located outside California^a.

^aSource : www.intel.com; les fautes d'orthographe figurent dans l'original.

28. Sur les origines du microprocesseur, voir www.intel.com/intel/intelismuseum/exhibit/hist_micro/hof/hof_main.htm et www.motorola.com/content/0,1037,115-110,00.html.

Moore la publia dans « Cramming more components into integrated circuits », *Electronics*, 19 avril 1965 [140]. Cet article devenu célèbre a introduit dans la conception des circuits intégrés une « anticipation autoréalisatrice », comme disent les économistes : ceux qui conçoivent un nouveau circuit se font un devoir de la respecter (Colwell [39]). Par ailleurs elle a encouragé les chercheurs à anticiper sur la croissance des performances et à concevoir des systèmes utilisant une puissance très supérieure à celle disponible lors de leurs recherches (Kay [98] p. 515), ce qui a suscité une accélération de l'innovation.

Voici la phrase clé de cet article : « Le niveau de complexité qui permet de minimiser le coût moyen par composant a doublé chaque année. À court terme, cette croissance va se poursuivre sinon s'accélérer. À long terme le taux de croissance est un peu moins certain, mais il n'y a pas de raison de penser qu'il ne restera pas à peu près constant pendant au moins dix ans. Cela signifie qu'en 1975 le nombre de composants correspondant au coût minimum sur un circuit intégré sera de 65 000²⁹. »

Le raisonnement de Moore comporte deux étapes. Il examine d'abord la relation entre le coût moyen de production par composant et le nombre de composants sur un circuit (qu'il appelle « complexité »). Cette fonction est d'abord décroissante, puis croissante : il existe donc un niveau de complexité pour lequel le coût moyen par composant intégré sur le circuit est minimal. C'est ce niveau que des producteurs rationnels choisiront, car il procure le meilleur rapport efficacité/coût. Ensuite, Moore constate que ce niveau optimal de complexité a été multiplié chaque année par deux.

Moore ne dit pas que le coût de production des circuits intégrés restera stable dans le temps malgré l'augmentation du nombre des composants : rien, dans son raisonnement, ne le garantit.

Le coût de production des circuits intégrés dépend peu de la quantité produite : c'est une *production à coût fixe*. Le coût de mise en production croît avec le niveau de complexité (la conception de l'Itanium a coûté à Intel 2 milliards de dollars³⁰ et, pour le mettre en production, il a fallu encore construire une usine coûteuse). Le coût moyen de production résulte de la division de ce coût fixe par le nombre d'unités vendues ; le prix de vente sera égal au coût moyen augmenté de la marge que le fournisseur peut s'attribuer puisqu'il est en situation de monopole (ou d'oligopole si l'on suppose que le client peut choisir entre Intel, Motorola, AMD etc.).

La stratégie des fabricants de microprocesseurs est donc subtile : ils doivent dégager assez de profit pour financer leur recherche et leur croissance, mais aussi pratiquer des prix assez bas pour que le débouché de leurs produits puisse croître. Or le marché des ordinateurs est capricieux : formé

29. « The complexity for minimum components costs has increased at a rate of roughly a factor of two per year. Certainly over the short term this rate can be expected to continue, if not to increase. Over the long term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years. That means by 1975, the number of components per integrated circuit for minimal cost will be 65,000. »

30. Source : www.forbes.com/2001/05/29/0529intel_print.html

de l'addition d'un marché de premier équipement et d'un marché de renouvellement, il est sensible à la mode dans sa première composante et à la conjoncture dans la deuxième³¹. Il faut cadencer l'innovation pour le relancer périodiquement. On peut vendre cher les premiers microprocesseurs d'une nouvelle série, puisqu'ils procurent un gain de compétitivité (physique ou médiatique, peu importe ici) aux ordinateurs qui les utilisent ; puis il faut baisser leur prix pour élargir leur pénétration. L'évolution du prix des microprocesseurs résulte de l'ensemble de ces phénomènes ; si sa modélisation comporte la loi de Moore, elle ne s'y résume pas.

Étapes de la production d'un microprocesseur

La production des cristaux de silicium, matière première de base, suit un processus chimique dont le coût de revient est minime : à partir d'un germe de cristal, la silice hautement purifiée croît en un long cylindre à l'intérieur d'un creuset à haute température. Puis ce cylindre est découpé en tranches fines par une scie au diamant. Les tranches de silicium sont ensuite soumises à des manipulations de type photographique qui gravent les couches incorporant la logique et la physique du circuit intégré.

1) Chaque tranche est nettoyée à l'acide, puis placée dans un four à haute température où sa surface est oxydée.

2) La tranche est recouverte d'un produit photosensible et le dessin du premier niveau du circuit est projeté à travers un masque par un faisceau de rayons ultraviolets. Puis les surfaces impressionnées par ce faisceau sont enlevées par de l'acide. Le processus est répété pour chaque niveau du circuit. La tranche est traitée avec des impuretés chimiques positives ou négatives qui créent les zones conductrices. Finalement, elle est revêtue d'un enduit qui protège sa surface et empêche les fuites de charges électriques.

3) Les puces comportant des fautes sont repérées sur la tranche par inspection visuelle et test informatique. Puis les puces sont découpées et collées sur un support. De minuscules fils sont soudés pour connecter le support aux points de contact de la puce, un couvercle est placé sur la puce et scellé pour la protéger.

4) Le microprocesseur est mis pendant plusieurs jours dans un four à basse température pour simuler son utilisation à long terme. Il est ensuite testé et les microprocesseurs à basse performance sont éliminés. (Source : Malone [124])

En 1975, Moore réévalua le rythme de croissance : désormais elle procédait par doublement *tous les 18 mois* et non tous les ans. Néanmoins elle

31. Tout marché de renouvellement (automobiles, équipement ménagers etc.) est le théâtre d'une amplification du mouvement conjoncturel, le consommateur retardant l'achat de quelques mois ou années en basse conjoncture, et achetant en masse en haute conjoncture (voir page 56).

restait exponentielle. Elle s'écrit donc, en notant n_t le nombre de composants sur une puce l'année t :

$$n_t = n_{1975} 2^{(t-1975)/1,5}$$

L'accroissement de la densité des composants permet d'augmenter les performances : lorsque la distance entre transistors se réduit la vitesse de traitement s'accroît et on peut introduire sur la puce des fonctions auparavant remplies par d'autres équipements comme la carte graphique, le modem ou le contrôle de la mémoire.

En 1995, Moore vérifia que la progression prévue avait bien été respectée. Cependant lors du forum des programmeurs d'Intel de septembre 1997 il a déclaré que l'accroissement de la densité des microprocesseurs pourrait atteindre vers 2017 une limite physique, celle de la taille des atomes.

L'évolution des performances que décrit la loi de Moore résulte de la mise en exploitation progressive d'un phénomène naturel. Lorsque la croissance s'arrêtera vers 2017 nous devrions, si la prolongation de la tendance se confirme, disposer de processeurs et de mémoires $2^{12/1,5} = 256$ fois plus puissants qu'en 2005. L'informatique sera alors qualitativement différente de celle que nous connaissons aujourd'hui.

Aurons-nous répondu à toutes les questions que pose l'utilisation de cette ressource ? C'est peu probable. Ces questions relevant de l'organisation des entreprises et de la vie en société, elles concernent tout le monde et ne pourront pas se régler aussi vite que ne le font les questions techniques qui, elles, sont traitées par la sous-population restreinte des ingénieurs « pointus ».

L'utilisation de la ressource naturelle que constitue le silicium, matière peu coûteuse mais riche en potentialités, nous occupera pendant le XXI^e siècle et sans doute encore par la suite.

1.5.3 Évolution du prix des micro-ordinateurs

L'indice du prix des micro-ordinateurs est l'un des indicateurs les plus importants pour la compréhension de l'économie des TIC. L'INSEE le publie sur www.insee.fr, série PVIC300201. Il est disponible selon la périodicité trimestrielle depuis le premier trimestre 1988, selon la périodicité mensuelle depuis janvier 2003. Nous présentons ici une série longue composée en enchaînant les bases successives.

Il s'agit d'un indice hédonique, c'est-à-dire à qualité constante. La moitié environ de la baisse de l'indice s'explique par la baisse du prix moyen de vente des micro-ordinateurs, l'autre moitié s'explique par la hausse de la qualité.

La courbe de la figure 1.11 prend à partir de 1990 l'allure d'une exponentielle décroissante ; la valeur de l'indice est en octobre 2005 égale à 0,6 % de sa valeur au premier trimestre 1988.

Pour pouvoir l'interpréter, observons le taux de variation en équivalent annuel (le niveau « - 20 % » sur la figure 1.12 signifie que l'indice a évolué ce trimestre-là à un taux qui, sur un an, correspondrait à une baisse de

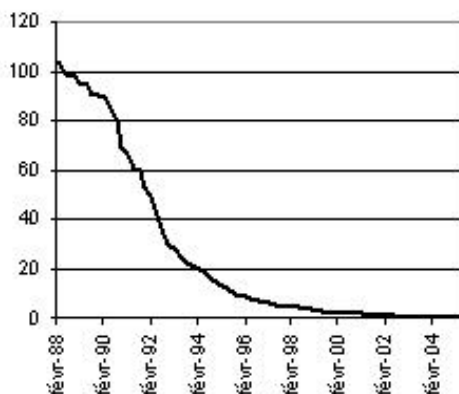


FIG. 1.11 – *Indice du prix de vente industriel des micro-ordinateurs, INSEE*

20 %). Le graphique représente l'évolution de la moyenne mobile sur cinq trimestres.

On voit que la baisse annuelle a été durant les années 90 de l'ordre de 30 %. La rapidité de cette baisse explique la pénétration rapide des micro-ordinateurs dans les entreprises et dans le grand public. La baisse s'est ralentie dans les années 2000 : elle se poursuit depuis la mi-2003 au rythme annuel de 8,2 %.

Le *Department of Labor* des États-Unis publie depuis décembre 1992 un indice mensuel du prix des ordinateurs de bureau et un autre pour les ordinateurs portables³². La comparaison des taux de variation (en moyenne mobile sur cinq trimestres) montre des évolutions parallèles, la baisse du prix des portables étant toutefois un peu plus rapide que celle des ordinateurs de bureau (figure 1.13).

En comparant les taux de variation des indices français et américains, on voit que si la chronologie a été différente l'ampleur de la baisse a été comparable dans les deux pays (figure 1.14).

On voit apparaître aux États-Unis une accélération de la baisse à partir de mars 2004 ; ce phénomène ne s'est pas, ou pas encore, produit en France.

1.6 Conjoncture des TIC

Un petit modèle permet d'éclairer la crise que connaissent les TIC (informatique, télécoms) depuis un retournement que l'on peut dater du 8 mars 2000, date du début de la chute du cours de l'action de France Telecom.

³². Source: Department of Labor « Producer Price Index », stats.bls.gov/, Séries WPU11510114 pour les « Personal Computers and Workstations », WPU11510115 pour les « Portable Computers ».

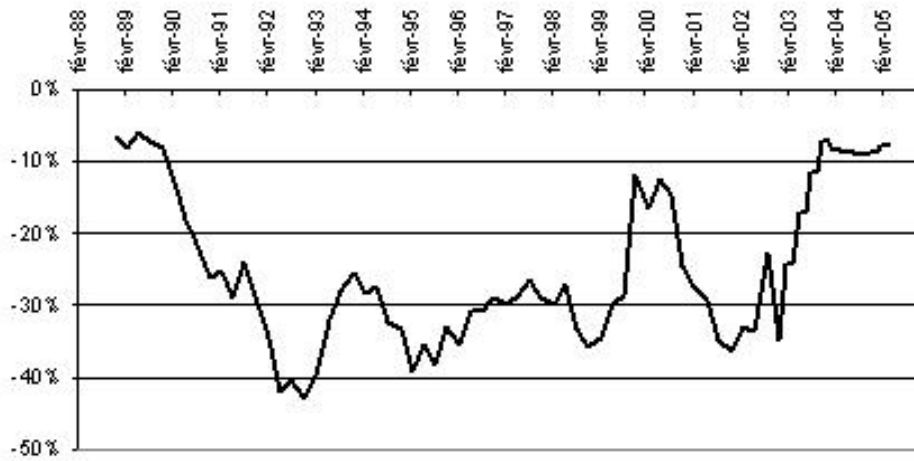


FIG. 1.12 – Taux de variation annuel du prix des micro-ordinateurs

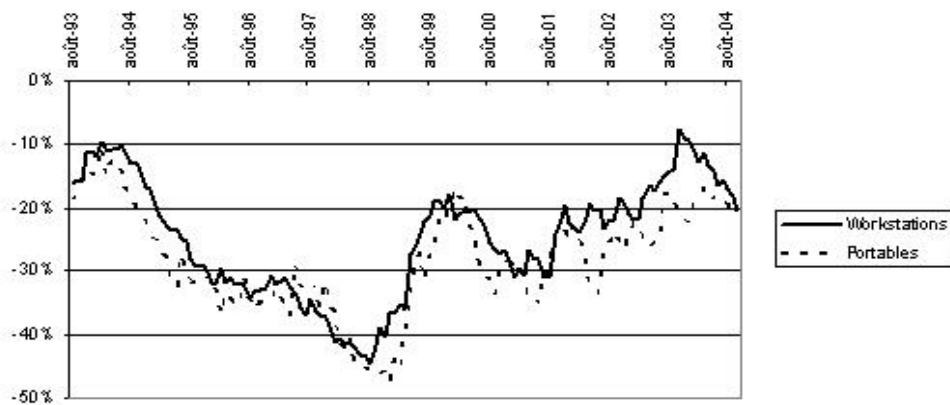


FIG. 1.13 – Taux de variation des prix des ordinateurs de bureau et des ordinateurs portables aux États-Unis

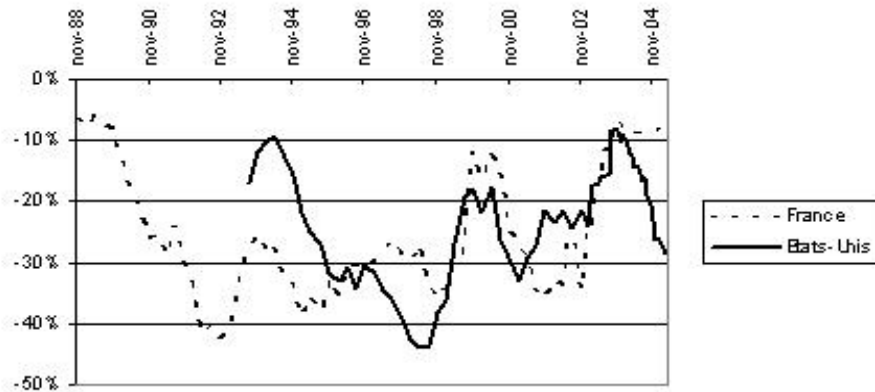


FIG. 1.14 – Taux de variation des prix français et américain des micro-ordinateurs

Il faut d'abord considérer les lois de la pénétration d'un produit innovant, que « les marchés » (c'est ainsi que l'on désigne la Bourse) ont semble-t-il oubliées lorsqu'ils interprétaient les statistiques. Puis, en examinant comment fonctionne la conjoncture des produits arrivés au stade du renouvellement, on peut anticiper les défis auxquels seront confrontés les fournisseurs.

Ce rapide parcours permet de voir que derrière la crise qui a débuté en mars 2000, due à un ralentissement prévisible et pourtant imprévu, se profile une crise plus grave provoquée par un changement de régime conjoncturel auquel les stratégies devront s'adapter.

* *

La pénétration d'un produit nouveau (bien ou service) dans la consommation suit une loi *logistique* : la courbe ressemble d'abord à une exponentielle croissante, puis s'infléchit ; sa croissance ralentit et elle s'aligne enfin sur la pénétration asymptotique. C'est ainsi qu'ont pénétré les produits électroménagers et les services de télécommunications.

La fonction logistique fait partie depuis longtemps de l'arsenal théorique du marketing. L'une de ses expressions est la loi de Gompertz :

$$y_t = \delta + \alpha e^{-e^{\beta - \gamma t}}$$

où y_t est le taux de pénétration. Le niveau asymptotique est $\alpha + \delta$. La figure 1.15 représente l'évolution de la consommation sur 30 ans en supposant $\alpha = 1$, $\beta = 2,5$, $\gamma = 0,3$, $\delta = 0$. Les paramètres ont été choisis de sorte que le point d'inflexion, situé au taux de pénétration 36,8 %, se trouve dans l'année 8.

Lorsqu'un produit est non seulement nouveau, mais innovant, c'est-à-dire lorsqu'il bouscule les usages courants, la logistique est pentue : le dé-

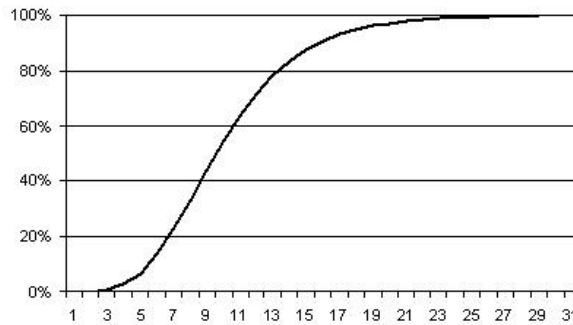


FIG. 1.15 – Loi logistique de pénétration d'un nouveau produit

marrage est plus lent (il faut vaincre des réticences), par contre la croissance est rapide une fois qu'elle s'est enclenchée³³ (« effet d'avalanche »). L'inflexion est plus lente à venir mais se produit tout de même : la prolongation illimitée d'une exponentielle est impossible, même en tenant compte d'un éventuel « multi-équipement » (personnes qui ont plusieurs automobiles, plusieurs téléviseurs, plusieurs ordinateurs, plusieurs téléphones etc.) et de la diversification des produits fournis sur la plate-forme technique une fois celle-ci installée : la consommation de ces produits ne peut pas excéder le niveau du PIB !

* *

Les « investisseurs » (c'est ainsi, bizarrement, que l'on appelle les personnes qui achètent des actions à la Bourse) regardent les comptes et les taux de croissance trimestriels. Ils sont moins attentifs aux « fondamentaux » dont fait partie le niveau asymptotique de la pénétration. Extrapolant la croissance initiale, ils anticipent une évolution exponentielle des chiffres d'affaire et des profits. Cette anticipation les incite à attribuer aux entreprises des TIC une valeur élevée. Ainsi valorisées, celles-ci peuvent se procurer des fonds en émettant des actions nouvelles ; leur actif, qui comprend des actions d'autres entreprises du secteur, est lui aussi fortement valorisé. Leur taux d'endettement paraît alors bas ; les banques, alléchées, leur proposent des prêts qui financeront l'achat d'autres entreprises (coup double pour la banque : elle prête à une entreprise solvable et elle encaisse une commission). Ainsi se met en place un piège.

Ce qui devait se produire finit par arriver : la pénétration s'infléchit, la croissance ralentit. Le dimensionnement des machines et des réseaux, le niveau des stocks d'équipements, préparés pour une demande supérieure, apparaissent alors excessifs (surcapitalisation).

³³. Dans le cas de l'économie des réseaux, on explique l'effet d'avalanche (que l'on nomme alors « effet de réseau ») par le fait que l'utilité d'un raccordement nouveau est fonction croissante du nombre des personnes déjà raccordées. Dans le cas des TIC, l'effet de réseau joue et aussi l'effet de mode (« il faut être sur l'Internet »).

Les investisseurs qui anticipaient une exponentielle se trouvent confrontés à une logistique. Leurs anticipations sont modifiées ainsi que leur évaluation des entreprises. Le cours des actions s'effondre. Les actifs se dévalorisant, les bilans sont mécaniquement détériorés. Le taux d'endettement effraie alors les banques qui, tremblant pour leurs créances, refusent désormais le renouvellement des prêts qui allait auparavant de soi. C'est la faillite. « Ils ne mouroient pas tous, mais tous étoient frappés³⁴ ».

Pourquoi cette épidémie? *parce que les investisseurs ont pris le début d'une logistique pour une exponentielle*. Ils ont été confortés dans cette erreur par des plus-values substantielles (la croissance exponentielle semblait se confirmer), par les indicateurs de court terme (qui ne montraient au début aucune tendance à l'inflexion), par la conviction immodeste que « les marchés ont toujours raison », par le manque d'attention envers les enseignements classiques du marketing.

Il est vrai qu'un autre phénomène a compliqué l'interprétation de la pénétration : la baisse rapide des prix (30 % par an à qualité constante, 15 % par an en prix moyen pour les micro-ordinateurs) a élargi le marché potentiel et fait monter l'asymptote de la pénétration. La croissance résultait donc de deux mouvements conjugués : une logistique qui se rapprochait de l'asymptote correspondant au prix courant ; une asymptote qui se déplaçait vers le haut en raison de la baisse du prix.

La liste des difficultés s'arrête là pour les opérateurs de télécommunications dont le revenu est récurrent : le nombre des abonnés peut cesser de croître ainsi que leur consommation, mais ils continuent de payer l'abonnement et de consommer le service³⁵. Pour les fournisseurs de composants et d'équipements par contre une autre difficulté se présente : l'évolution apporte un changement de régime, car ils passent d'un marché de premier équipement à un marché de renouvellement dont la conjoncture est beaucoup plus accidentée.

* *

Le marché des téléphones mobiles, des ordinateurs, est l'addition de deux marchés : un marché du premier équipement gouverné par la logistique de pénétration que nous venons d'examiner, et un marché de renouvellement dont la part devient de plus en plus importante à mesure que la pénétration croît.

Il existe de purs marchés de renouvellement comme celui de l'automobile. Elle a depuis longtemps atteint sa pénétration asymptotique dans les pays riches : les acheteurs de voiture neuve sont pour la plupart d'anciens propriétaires de voiture qui ont voulu acheter un nouveau modèle pour bénéficier d'un accroissement de confort et peut-être de prestige social. La durée de possession d'une même automobile est ainsi de l'ordre de cinq à sept ans³⁶.

34. Jean de La Fontaine, « Les animaux malades de la peste », *Fables*.

35. Toutefois la téléphonie sur IP est de nature à compromettre le revenu récurrent lui-même.

36. Il n'en est pas de même du marché des réfrigérateurs et des machines à laver, qui

De même, la durée de vie économique d'un micro-ordinateur, d'un téléphone mobile, est de quelques années. Après ce délai, ces matériels fonctionnent encore mais ils sont « dépassés » par des innovations survenues entre temps et qui font envie aux utilisateurs : ils sont devenus *obsoletes*.

L'effet de mode, très fort pour les micro-ordinateurs et les téléphones mobiles, rapproche ces produits de l'automobile mais avec une durée de vie économique plus courte encore et un marché de l'occasion pratiquement inexistant, ce qui donne encore plus de poids au phénomène du renouvellement.

L'étude de l'industrie automobile illustre le fait que *le marché des biens durables est un amplificateur de la conjoncture générale*. En effet lorsque la conjoncture est défavorable le remplacement d'une automobile peut être reporté à des jours meilleurs. En revanche lorsqu'elle est favorable les clients se précipitent pour acheter. On représente ces comportements par une probabilité de remplacement croissant d'autant plus vite avec l'âge du véhicule que la conjoncture est meilleure. Le renouvellement sera fonction de la pyramide des âges et de la conjoncture, et par ailleurs il modifie la pyramide des âges. La rétroaction que comporte cette dynamique amplifie les fluctuations conjoncturelles.

On peut en tirer une leçon : dès que la part du renouvellement dans le marché des TIC est importante, *la conjoncture du marché des composants et des équipements devient un amplificateur de la conjoncture générale*. Cela posera aux entreprises des TIC un défi auquel rien ne les a préparées.

* *

Pour illustrer le scénario ci-dessus, nous construirons un petit modèle. Nous négligeons l'effet de la baisse tendancielle du prix des TIC : elle fait monter l'asymptote de la pénétration, mais ce phénomène devient de plus en plus faible lorsque la baisse du prix se poursuit. Supposons que la croissance tendancielle du PIB soit de 3 % par an et qu'il subisse des oscillations conjoncturelles d'une périodicité de cinq ans. Le PIB de l'année t est alors donné par l'expression

$$PIB_t = PIB_0[1 + a \cdot \sin(\omega t)]e^{rt}$$

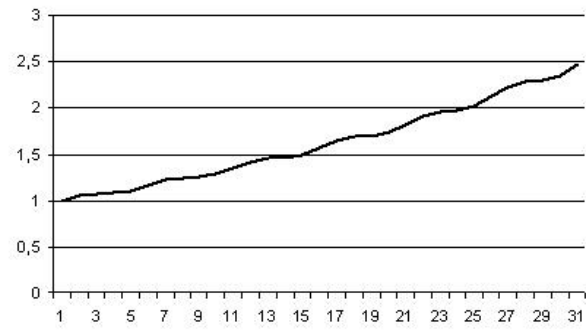
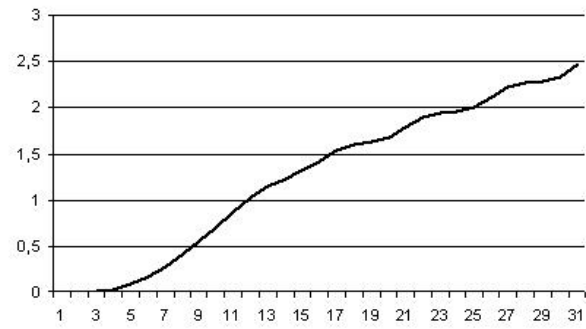
avec $r = 0,03$ et $\omega = 2\pi/5$. Le graphique 1.16 couvre 30 ans ; nous avons supposé que $a = 0,02$.

Supposons que la pénétration suive la loi de Gompertz évoquée ci-dessus et que le niveau asymptotique de cette loi évolue comme le PIB ; le parc installé lors de l'année t est obtenu en combinant les deux relations précédentes :

$$Parc_t = y_t PIB_t$$

On remarque sur le graphique 1.17 que l'influence des oscillations conjoncturelles sur le niveau du parc est peu sensible dans la phase de pénétration

engagent moins le prestige social : on ne remplace ces machines que lorsqu'elles sont en panne et leur durée de vie est de l'ordre de vingt ans. Le marché du premier équipement (équipement des logements neufs) reste donc significatif pour ces produits.

FIG. 1.16 – *Croissance tendancielle et fluctuations conjoncturelles*FIG. 1.17 – *Evolution du parc installé*

initiale. Elle devient de plus en plus sensible par la suite.

Le renouvellement du parc se fait de façon différente selon que l'économie se trouve en haut ou en bas du cycle conjoncturel. Nous supposons que les probabilités P_A de renouvellement du matériel en fonction de l'âge A sont celles données par le tableau 1.1.

Année	0	1	2	3	4	5	6	7	8	9
Haut de cycle	0	0,2	0,4	0,7	0,9	1	1	1	1	1
Bas de cycle	0	0,1	0,2	0,35	0,45	0,5	0,6	0,7	0,9	1

TAB. 1.1 – Probabilité de renouvellement en fonction de l'âge du matériel

Les distributions des durées de vie qui correspondent à ces probabilités sont représentées par le graphique 1.18, la probabilité de la durée de vie D étant :

$$Prob_D = P_D \prod_{A=0}^{D-1} (1 - P_A)$$

Si la conjoncture restait durablement en haut de cycle, la durée de vie moyenne $\sum_{D=0}^{\infty} D Prob_D$ serait de 2,78 ans; en bas de cycle, elle serait de 3,89 ans.

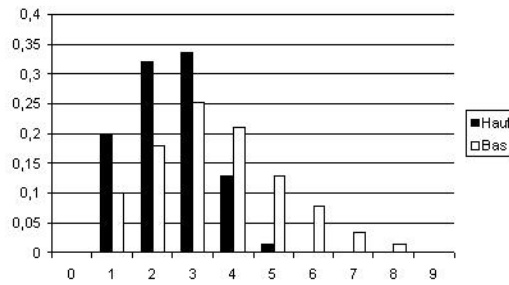


FIG. 1.18 – Distributions des durées de vie

Nous supposons que dans les situations intermédiaires la probabilité de renouvellement est une moyenne pondérée des deux valeurs extrêmes. Notons $\varphi(t)$ la fonction qui indique à quel point du cycle se trouve l'économie :

$$\varphi(t) = [1 + \sin(\omega t)]/2$$

La probabilité du renouvellement d'un ordinateur d'âge A lors de l'année t est alors, en notant respectivement P_{AH} et P_{AB} les probabilités du renouvellement en haut et en bas de cycle :

$$P_{A,t} = \varphi(t)P_{AH} + [1 - \varphi(t)]P_{AB}$$

La production lors de l'année t doit permettre l'accroissement annuel du parc et le renouvellement des ordinateurs obsolètes soit, en notant $Parc_{A,t}$ l'effectif du parc d'âge A lors de l'année t :

$$Prod_t = Parc_t - Parc_{t-1} + \sum_A P_{A,t} Parc_{A-1,t-1}$$

L'évolution de la production a alors l'allure qu'indique le graphique 1.19.

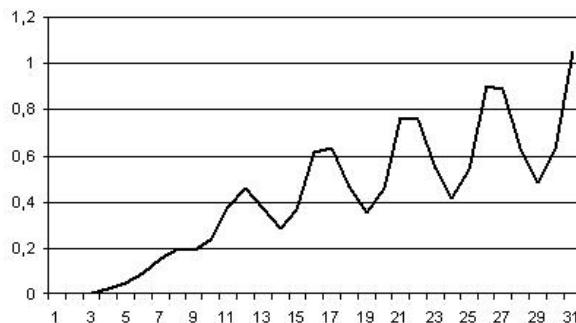


FIG. 1.19 – Évolution de la production

Ce graphique effrayant fait apparaître, lorsque la pénétration est assez forte pour que le renouvellement soit significatif, une succession de hausses et de baisses importantes de la production. Il illustre l'amplification de la conjoncture que suscite l'alternance des échéanciers de renouvellement : la production est *beaucoup plus accidentée* que la croissance du parc.

Les entreprises du secteur des TIC sont en train de subir un ralentissement qui révèle un surdimensionnement, le secteur s'étant préparé à la poursuite illimitée d'une croissance exponentielle. Il s'ensuit une dévalorisation des actifs physiques (et pas seulement des actifs boursiers), un effondrement des commandes passées aux fournisseurs et équipementiers (ce qui suscite chez eux une crise encore plus grave), bref un « coup de frein » dont les conséquences s'amplifient chez les fournisseurs de la même façon qu'un ralentissement sur une autoroute se traduit en amont par un bouchon où les véhicules sont à l'arrêt.

La dévalorisation des actifs va provoquer des disparitions d'entreprises jusqu'à ce que le niveau des actifs redevienne compatible avec les perspectives de croissance révisées.

Après ce choc, qui résulte de la révision des anticipations et de la prise de conscience que la pénétration est non pas exponentielle mais logistique, se profile pour les fournisseurs de composants et d'équipements un second choc encore plus violent : l'adaptation à un marché de renouvellement.

Les entreprises des TIC, conquérantes et orgueilleuses (d'autant plus que leurs dirigeants, ayant été les premiers à se placer sur un marché à fort potentiel, ont bénéficié d'une réputation de « génie »), vont devoir

devenir modestes, adopter les stratégies prudentes de l'industrie automobile, pratiquer un marketing attentif à des fluctuations de la demande qu'il leur faudra savoir anticiper.

Cette adaptation sera culturellement difficile pour les dirigeants du secteur des TIC. Elle risque de susciter une nouvelle vague de faillites qui sera due cette fois non à la dévalorisation des actifs, mais à la dévalorisation de la stratégie.

Chapitre 2

Automatisme et intelligence

« Ne tournons pas nos regards vers l'Amérique pour copier servilement les institutions qu'elle s'est données, mais pour mieux comprendre celles qui nous conviennent, moins pour y puiser des exemples que des enseignements, pour lui emprunter les principes plutôt que les détails de ses lois. » (Alexis de Tocqueville, avertissement de la douzième édition de *De la démocratie en Amérique*, [204] p. XLIV).

L'histoire de l'informatique, qui s'amorce dans les années 1940, s'éclaire si on la situe dans la perspective de l'industrialisation du travail de bureau dont on peut situer le début vers 1880 à Chicago. D'abord utilisée pour accroître la productivité du travail en automatisant les tâches répétitives, l'informatique s'est enrichie dans les années 1990 des apports de la bureautique communicante. Elle équipe désormais les processus de production au plus près de leur déroulement. L'entreprise, entrée dans l'ère du *travail assisté par ordinateur*, doit alors s'efforcer d'articuler au mieux deux démarches : l'organisation de l'être humain et la programmation de l'automate. Cette tâche n'est pas aisée car l'ordre n'est pas l'état naturel d'un système d'information, dont la qualité est sapée par une entropie toujours renaissante.

* *

Même si la contribution des Européens à l'informatique a été importante, ses progrès essentiels ont tous été réalisés aux États-Unis : l'informatique étant à la fine pointe de l'organisation de l'entreprise, elle est toute naturelle pour des Américains dont le pays a, dès sa formation, attribué à l'Entreprise l'hégémonie culturelle qui en Europe appartient à l'État (Gramsci [72] p. 476).

Pour réussir l'articulation du travail humain et de l'automate, il faut percevoir la différence entre l'ordinateur et le cerveau humain : on ne peut pas en effet articuler deux êtres dont on aurait postulé l'identité. Cette question a été obscurcie par les controverses sur l'« intelligence de l'ordinateur », controverses qui s'éclairent si on les situe sur l'arrière-plan culturel

des États-Unis : est-il possible de planifier l'action au point que l'être humain, pour pouvoir être efficace, doive se faire l'exécutant d'un automate pré-programmé ? Ou bien l'action, avec ses incertitudes, nécessite-t-elle que l'on préserve la capacité de synthèse, de compréhension et de décision qui est propre au cerveau humain ?

* *

2.1 L'informatisation de l'entreprise

Pour décrire l'évolution historique du rôle de l'informatique, nous allons d'abord présenter de façon schématique comment une entreprise fonctionne, puis montrer comment l'informatique a progressivement équipé les divers types de fonctions. Le concept fondamental est celui d'*activité* que représente la figure 2.1.

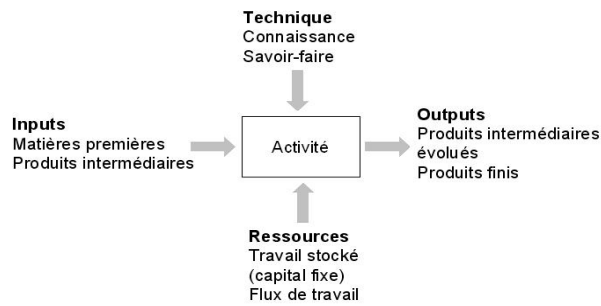


FIG. 2.1 – *Le concept fondamental : l'activité*

Toute entreprise, en effet, consomme des *inputs*, utilise des techniques et des ressources et produit des *outputs*. Donnons-en l'image la plus simple qui soit : à partir de planches et de clous, et en utilisant la technique du coup de marteau, on peut produire des caisses.

Si l'on veut analyser le fonctionnement de l'entreprise, toutefois, on ne peut pas en rester à cette image globale : il faut utiliser un « grain de photo » plus fin et identifier des activités élémentaires, la finesse du « grain » dépendant de l'objectif de l'analyse. Il apparaît alors que ces activités élémentaires s'enchaînent pour s'organiser en « processus de production ».

Tout processus débute par des événements externes à la production (ou « événements externes » tout court) : commandes des clients, livraisons des fournisseurs, réclamations etc. Lorsqu'il est initialisé par une commande d'un client, par exemple, le cycle du processus se boucle par une livraison accompagnée d'une facturation qui sera suivie d'un paiement (figure 2.2).

Arrêtons-nous un instant sur ce graphique qui résume la physique de l'entreprise et guide son informatisation. Les deux flèches à gauche représentent l'échange entre l'entreprise et le monde extérieur : elle reçoit des commandes et livre des *produits*. La boucle à droite représente le *processus*

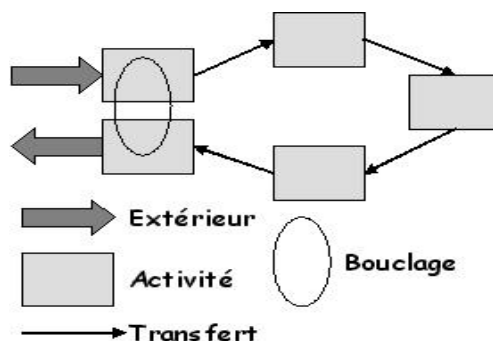


FIG. 2.2 – Bouclage du processus de production

de production interne à l'entreprise, avec ses étapes ou *activités*. Représenter le processus par une boucle, et non par une succession linéaire d'activités, souligne l'importance du *bouclage* qui permet le contrôle du délai et de la qualité de la production.

Ce schéma s'applique à l'entreprise, qu'elle soit informatisée ou non : des « sous-processus » font progresser le processus en lui fournissant des « livrables », produits intermédiaires documentaires ou physiques dont la mise à disposition est un « événement interne » (voir la figure 9.3 page 313).

2.1.1 Organisation du travail de bureau : années 1880

L'organisation du travail de bureau a résulté d'un effort prolongé et méthodique. Les progrès essentiels ont été réalisés lors des dernières décennies du XIX^e siècle dans le « loop » de Chicago, centre d'affaires en croissance rapide où furent mises au point les méthodes de standardisation et de classement documentaire ainsi que l'architecture des grands immeubles de bureau.

C'est également aux États-Unis que débuta la mécanisation du travail de bureau avec les machines à écrire et les machines à calculer : la machine à écrire a permis d'obtenir des documents plus lisibles que les manuscrits et de les dupliquer en quelques exemplaires grâce au papier carbone puis au stencil ; la machine à calculer a facilité les opérations de vérification et de calcul.

Les tâches de bureau remplies par les employés dans la première moitié du XX^e siècle se classent en deux catégories : celles internes à l'entreprise (*middle office* et *back office*), qui peuvent être entièrement organisées ; celles effectuées au contact des clients ou des fournisseurs, qui impliquent une part de dialogue et de négociation (« première ligne » ou *front office*).

* *

Les tâches du *back office* obéissaient toutes à un même schéma :

1) d'une part l'agent reçoit des commandes, des matières premières ou des produits intermédiaires ; son travail consiste à élaborer d'autres produits

intermédiaires (« livrables ») qu'il oriente vers l'étape suivante du processus. Par exemple, pour les agents qui traitaient l'information dans les grandes banques ou compagnies d'assurance, le travail se faisait sur un bureau dans une salle où se trouvaient de nombreux employés ; à gauche de l'agent se trouvait la barquette arrivée, à droite la barquette départ, les dossiers étant apportés et emportés par des personnes équipées de caddies (figure 2.3).

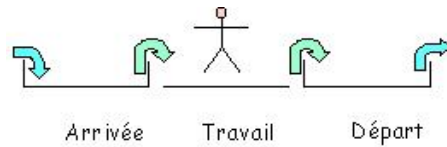


FIG. 2.3 – Poste de travail du back office

2) d'autre part les personnes qui transportent les dossiers d'un bureau à l'autre, ainsi que le superviseur de la salle de travail, assurent une logistique (*middle office*) qui entoure les tâches des agents d'un réseau de communication et de contrôle. La mesure du flux quotidien permet d'établir des normes de productivité. Les délais normaux de traitement d'une affaire peuvent être évalués. La pile de dossiers qui reste dans la barquette arrivée signale l'agent qui travaille plus lentement que les autres.

Évolution des équipements de bureau

Les équipements du travail de bureau (fauteuils, bureaux, téléphones, photocopieurs, télécopieurs, calculateurs, machines à écrire, classeurs, trombones, post-its, sans même évoquer l'ordinateur et sa séquelle d'imprimantes, scanners etc.) sont tous d'origine récente : le brevet du trombone est déposé en 1901, celui du classeur mécanique en 1904. Les copieurs apparaissent en 1890, mais la photocopie ne se répandra vraiment qu'à partir de 1960 avec la Xérographie. Le Post-it^a sera lancé par 3M (après de longues hésitations) en 1980.

La machine à écrire (D. Rehr [173]), inventée en 1868 par l'Américain Christopher Latham Sholes, est commercialisée par Remington en 1874. Elle a déjà le clavier QWERTY mais elle écrit en majuscules et l'auteur ne peut pas voir le texte qu'il tape. 5 000 machines sont vendues en cinq ans. La Remington n° 2 de 1878 permet d'écrire en minuscules et majuscules. En 1895, Underwood commercialise une machine qui permet de voir ce que l'on tape. Dès lors la machine à écrire se répand rapidement dans les entreprises. La première école de dactylographie est créée en 1911.

^a www.3m.com/about3M/pioneers/fry.html

Le travail que l'agent effectue sur un dossier consiste en calculs, vérifications et transcriptions, et aussi en expertises, classements, évaluations et décisions (ou formulation d'avis pour préparer les décisions). En même temps qu'il fait progresser le processus de traitement des affaires, ce travail alimente des fichiers manuels qui constituent la mémoire de l'entreprise. Les éventuelles interrogations donnent occasion à des échanges de notes ou de fiches que l'agent place dans la barquette départ en mentionnant le nom du destinataire, les réponses parvenant dans la barquette arrivée avec les dossiers à traiter.

Dans les entreprises industrielles, le travail de bureau traitait les commandes, les factures et la comptabilité; il émettait les ordres qui déclenchaient les opérations physiques de production, approvisionnement, stockage, transport et livraison. Les décisions concernant les opérations physiques étaient prises dans les bureaux, les décisions laissées aux agents de terrain étant celles qui accompagnent l'exécution.

Processus de l'entreprise industrielle

Dans l'entreprise industrielle, les commandes sont satisfaites en puisant dans les stocks; le suivi statistique du flux de commandes permet d'évaluer la demande anticipée et de déterminer le programme de production; les facteurs de production (capital K , travail L , biens intermédiaires X) sont mobilisés chacun selon le cycle de vie qui lui est propre; la fonction de production $Y = f(K, L, X)$ est mise en œuvre pour alimenter les stocks (figure 2.4).

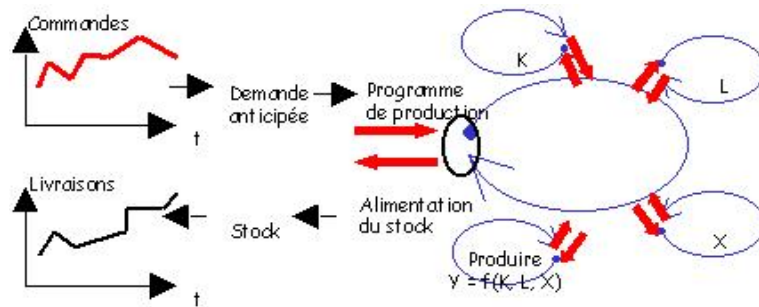


FIG. 2.4 – Anticipation et stockage dans l'entreprise industrielle

La procédure du *front office* était plus souple (figure 2.5), car il travaillait au contact d'un client ou d'un fournisseur, que ce soit en face-à-face, par téléphone ou par courrier: il ne s'agissait plus de traiter des documents conformes aux formats types de l'entreprise mais de répondre à des demandes ou questions formulées dans la langue de personnes extérieures à

l'entreprise et selon un ordre correspondant à leurs priorités. L'agent du *front office* devait transcrire les indications recueillies lors de la relation externe en un document susceptible d'alimenter le processus interne.

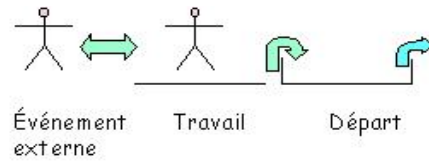


FIG. 2.5 – Poste de travail du front office

Cette organisation comportait des articulations fragiles. Les documents posés en pile risquaient d'être traités sur le mode LIFO (*last in, first out*) qui induit des délais aléatoires ; la succession des transferts entre agents pouvait finir « dans les sables » en cas d'erreur d'aiguillage ; si l'on avait besoin de retrouver un dossier en cours de traitement, il n'était pas facile de le suivre à la trace le long de son parcours. Enfin, le schéma que nous avons décrit se dégradait en variantes artisanales dans les entreprises petites et moyennes et il restait vulnérable à la négligence ou à l'étourderie.

2.1.2 Arrivée de l'informatique : années 1950

L'industrialisation du travail de bureau, avec les armoires de dossiers suspendus, classeurs, bibliothèques tournantes, la logistique du transport des dossiers, les longues opérations de calcul, appelait l'informatique. Mais l'informatisation n'a pris son essor que dans les années 1950, la guerre ayant pendant dix ans bloqué l'utilisation civile des techniques tout en accélérant leur conception¹.

La mécanographie, fondée sur le traitement électromécanique de cartes perforées par des trieuses et tabulatrices, a été d'abord conçue pour réaliser des travaux statistiques. La première réalisation est celle du statisticien américain Herman Hollerith (1860-1929) pour le recensement de la population des États-Unis en 1890. Les entreprises créées par Hollerith sont à l'origine d'IBM².

1. Comme ce fut le cas pour l'agriculture : en Europe le tracteur ne se répandra pas avant les années 1950.

2. Voir Donald E. Knuth [105] volume 3 p. 383. Hollerith fonde en 1896 la « Tabulating Machine Company ». Cette compagnie fusionne en 1911 avec la « Computing Scale Company of America » (balances automatiques) et l'« International Time Recording Company » (horloges enregistreuses) pour former la « Computing Tabulating Recording Company » (CTR) dont la direction est confiée à Thomas J. Watson (1874-1956). Ce dernier, devant le potentiel de la mécanographie pour la gestion, change en 1917 le nom de la filiale canadienne de la CTR en « International Business Machines » (IBM). La CTR adopte elle-même le nom d'IBM en 1924.

Les premiers utilisateurs de la mécanographie furent les instituts statistiques, les armées et quelques administrations (Carmille [31]). Les origines de plusieurs grands groupes informatiques remontent à l'ère de la mécanographie³.

C'est avec l'ordinateur, plus puissant que la machine mécanographique et surtout plus souple grâce à la mise en œuvre automatique de programmes enregistrés conformément à l'architecture de von Neumann [146] que l'informatique a pénétré les entreprises dans les années 1950 et surtout 1960. Elle a été d'abord utilisée pour automatiser la production physique : dès 1968, on a pensé à la commande numérique des machine-outils. Dans le numéro spécial de *Science et Vie* sur l'automatisme en 1964 la gestion n'apparaît encore que comme un domaine relativement secondaire pour l'automatisation.

Partage du travail entre l'ordinateur et l'être humain

Les entreprises achètent les ordinateurs pour économiser le temps que les agents opérationnels passent à des opérations répétitives de vérification, calcul et transcription, et aussi pour obtenir plus rapidement des informations de gestion d'une meilleure qualité.

Elles utilisent donc la machine pour faire des traitements (puissance) ainsi que pour classer et trier les données (mémoire).

Elles réservent à l'être humain les fonctions où il est supérieur à l'ordinateur : comprendre, expliquer, décider, concevoir.

Les premières entreprises de service à s'informatiser furent les banques et assurances ; dans les autres secteurs, les premières utilisations ont concerné la comptabilité, la paie et la gestion des stocks. Cela a modifié les conditions *physiques* du travail : les employés passaient dans les années 1960 une partie de leur temps à perforer des cartes et dépouiller des « listings » ; puis dans les années 1970 et 80 on installa des terminaux qui seront dans les années 1990 remplacés par des micro-ordinateurs en réseau. À chaque étape, l'ergonomie s'est modifiée ainsi que les possibilités offertes à l'utilisateur.

L'espace de travail change alors d'allure. Même si le bureau sans papier reste rare, les archives et dossiers sur papier sont remplacés, dans une large mesure, par des données stockées dans les mémoires électroniques. L'interface avec écran, clavier et souris s'installe sur tous les bureaux. Une part croissante du travail à faire arrive non plus dans une barquette, mais sur l'écran *via* le réseau.

3. Le bureau du Census demanda à James Powers de fabriquer les machines pour le recensement de 1910. Powers fonda en 1911 l'« Accounting and Tabulating Machine Company » : elle fusionna avec la « Remington Typewriter Company », présidée par James Rand, pour devenir en 1927 la « Remington Rand » qui fusionnera en 1955 avec « Sperry Gyroscope » pour former « Sperry Rand ». Frederick Bull (1882-1925), ingénieur norvégien, créa en 1922 avec son associé Kurt Kruesen une société de fabrication de machines mécanographiques. Les brevets furent achetés en 1929 par le groupe suisse Egli qui fonda la compagnie « Egli-Bull ». En 1932 le groupe français Caillies racheta les brevets au groupe « Egli-Bull » pour créer en 1933 la « Société des Machines Bull ».

Décalage de la pénétration des innovations

Lorsqu'on examine comment l'informatique a pénétré les entreprises, on constate un délai important entre la disponibilité d'une innovation et sa mise en œuvre. La chronologie des innovations est donc différente de celle de leur utilisation.

Ainsi, il était dès 1957 possible d'utiliser quatre terminaux en grappe sur l'IBM 305 ; mais les entreprises en sont restées pendant les années 1960 au couple « carte perforée et listing » et la diffusion des terminaux date des années 1970. De même, il était dès le début des années 1980 possible de fournir aux utilisateurs des micro-ordinateurs en réseau ; mais de nombreuses entreprises ont continué à utiliser des terminaux passifs jusqu'au milieu des années 1990.

Ces décalages s'expliquent : la première version d'une solution innovante est coûteuse et demande des mises au point, son utilisation implique des changements également coûteux de l'organisation. L'entreprise prendra donc tout son temps avant de comprendre l'utilité d'une innovation, puis de la mettre en œuvre.

Ce changement ne modifie pas fondamentalement la nature du travail : la différence entre événement interne et événement externe reste de même nature, même si l'écran-clavier s'impose désormais comme un tiers dans la relation avec les personnes extérieures à l'entreprise, au point parfois de gêner le dialogue (figure 2.6).

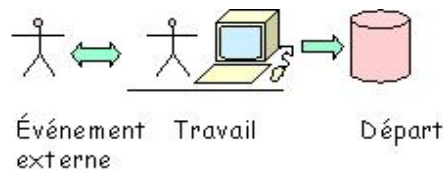


FIG. 2.6 – L'informatisation du front office

Toutefois l'agent n'a plus, en principe, à recopier une information déjà introduite dans l'ordinateur ; la vérification de la saisie peut être automatique ; les calculs (de comptes, prix, taxes, salaires, ainsi que les totalisations etc.) sont eux aussi automatisés, ainsi que la mise en forme et l'édition des divers « états » (bulletins de paie, documents comptables, état des stocks, statistiques etc.).

L'ordinateur remplit ainsi deux fonctions : d'une part il aide à traiter des dossiers individuels dont il facilite aussi le tri et la recherche ; d'autre part il permet de produire des indicateurs. L'être humain pour sa part se spécialise dans les tâches qu'il fait mieux que l'ordinateur : il analyse l'information pour faire le tour d'un problème, l'interprète pour comprendre, la

synthétise pour résumer et communiquer ce qu'il a compris ; enfin il décide ou même il conçoit. Ayant été soulagé des travaux qui exigeaient une utilisation répétitive de son cerveau, il est invité à se consacrer à ceux auxquels cet organe est le mieux adapté. On arrive ainsi à un partage des tâches où chacune des deux ressources (le « silicium », la « matière grise ») tend à être utilisée au mieux de ses aptitudes. Toutefois cette évolution n'est pas facile : elle est pénible pour ceux des agents, parfois les plus intelligents, qui avaient pris l'habitude de travailler de façon mécanique et rapide tout en pensant à autre chose. Désormais le travail leur demande concentration, réflexion, responsabilité, prise de risque. Même s'il est en principe devenu plus intéressant, il implique un effort psychologique accru.

Il faut aussi des changements dans l'organisation (transversalité etc.) : l'entreprise, qui doit accorder à l'employé un pouvoir de décision correspondant aux responsabilités qu'elle lui confie, doit aussi éviter de le harceler pour obtenir toujours plus de productivité, de qualité ou de profit unitaire. Les rapports entre personnes doivent devenir *respectueux* : quand la décision est décentralisée, il faut en effet savoir *écouter ce que dit l'autre*.

Les entreprises sont cependant parfois tentées d'oublier que l'on ne peut pas demander à un salarié d'être à la fois un exécutant docile et un pionnier plein d'initiative et de créativité.

* *

Nous sommes là vers le milieu des années 1980. Il faut compléter cette description en mentionnant des défauts souvent rencontrés. D'une part les « applications » informatiques ont été conçues séparément et communiquent mal : les agents doivent dans le cours d'une même tâche ouvrir une session, puis la fermer pour passer à une autre dont l'ergonomie sera différente, ressaisir des données, utiliser des codes divers dont la mémorisation demande un apprentissage. Si l'informatique est puissante, elle manque encore de cohérence et de « convivialité ».

C'est que l'automate n'est pas souple, et comme il ne s'adapte pas facilement aux utilisateurs l'entreprise leur demande de s'adapter à lui. Ses défauts sont d'abord tolérés en raison des gains d'efficacité que l'informatique apporte mais ils deviennent de plus en plus insupportables.

Le « système d'information » vise à les corriger. Les diverses applications doivent s'appuyer sur un référentiel unique, ce qui garantit leur cohérence sémantique ; elles doivent échanger les données et se tenir à jour mutuellement, ce qui assure la cohérence de leur contenu et supprime les ressaisies.

Toutefois cette mise en ordre reste souvent partielle et les défauts persistent en raison du poids de l'existant et de la pression d'autres priorités.

2.1.3 La bureautique communicante : années 1980

L'arrivée du micro-ordinateur dans les années 1980 (Penny et Volle [216]) fut un choc pour les informaticiens qui ne reconnurent pas immédiatement sa légitimité ni son utilité. Le micro-ordinateur servit d'abord à diffuser des applications de bureautique personnelle qui avaient été mises au point

sur des architectures de mini-ordinateurs en grappe (traitement de texte, tableur, grapheur). Il supplanta ainsi progressivement la machine à écrire et la machine à calculer.

Cependant les applications bureautiques se sont d'abord déployées dans le désordre (versions différentes des applications, travaux locaux sans cohérence d'ensemble). Au début des années 1990 la mise en réseau des micro-ordinateurs a confronté la bureautique aux exigences de cohérence du système d'information : pour toute donnée importante, seule doit exister sur le réseau une mesure définie et tenue à jour par le propriétaire de la donnée.

Le micro-ordinateur a finalement cumulé plusieurs rôles : d'une part il remplace les terminaux pour l'accès aux applications centrales, d'autre part il apporte à l'utilisateur la bureautique personnelle, puis la « bureautique communicante » (messagerie, documentation électronique, « groupware » puis Intranet). Le PC en réseau devient ainsi tout à la fois le terminal ergonomique des applications centrales, un outil de communication asynchrone entre agents et la porte d'accès aux ressources documentaires de l'entreprise (figure 2.7).

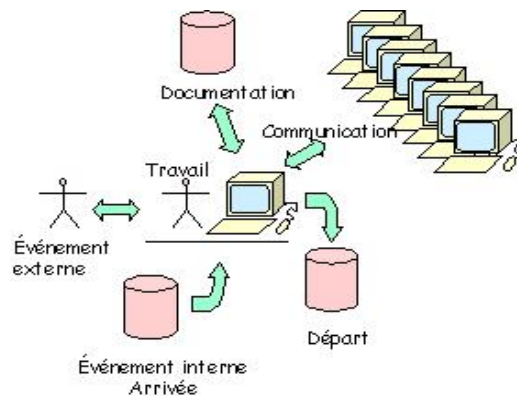


FIG. 2.7 – Poste de travail informatisé

On dirait alors que l'informatique a accompli tout ce qui lui était possible : elle fournit à l'utilisateur une interface qui, fédérant sous une ergonomie cohérente les accès aux diverses applications, lui évite les connexions-déconnexions et les doubles saisies tout en soulageant son effort de mémoire ; elle lui fournit aussi un moyen de communication, un *média*. Il lui reste cependant à assister les utilisateurs non seulement dans chacune de leurs tâches considérée séparément, mais dans la succession et l'articulation des activités tout au long du processus de production.

En effet si l'informatique a libéré l'agent opérationnel des tâches répétitives de calcul, vérification et transcription, les entreprises ne l'ont pas encore pleinement utilisée pour assurer les fonctions de logistique et de supervision, de *middle office*, remplies autrefois par les personnes qui transportaient les dossiers et par les superviseurs des salles de travail. Or le travail, devenu informatique (« virtuel »), a perdu la visibilité que lui conférait

l'apparence physique des documents et dossiers sur papier. Il est devenu plus difficile de vérifier sa qualité, d'évaluer la productivité des agents et de maîtriser les délais de production.

Rien de tout cela n'est cependant impossible pour l'informatique. Les outils existent depuis longtemps (de premiers *workflows* ont fonctionné dès l'époque des *mainframes*), mais pour qu'ils soient mis en œuvre il faut que le besoin soit ressenti et que la possibilité de le satisfaire soit perçue. L'attention s'était d'abord focalisée sur la productivité de l'agent individuel ainsi que sur la maîtrise des *concepts* (composants, classes, attributs, fonctions) que l'informatique mettait à sa disposition. Il fallait maintenant utiliser celle-ci pour automatiser les *processus* de production eux-mêmes - ou, plus précisément, pour apporter l'assistance de l'automate aux agents opérationnels chargés de ces processus.

L'informatique communicante apporte un élément de solution : s'il est possible aux utilisateurs de communiquer par messagerie, pourquoi ne pas utiliser ce média pour tisser une solidarité entre les étapes d'un même processus ?

2.1.4 Du concept au processus : années 1990

Pour retrouver la maîtrise de la logistique que l'informatique avait dans un premier temps négligée, il fallait introduire dans le système d'information les tables d'adressage qui balisent les transferts entre agents successifs, la traçabilité (possibilité de retrouver et consulter un dossier en cours de traitement), des indicateurs de volume, de délai et de qualité : ce sont là les fonctionnalités du *workflow*⁴. Il améliore notablement la logistique par rapport à l'époque du papier : il supprime le risque du *last in, first out*, assure la traçabilité et produit automatiquement des indicateurs de volume et de délai qui facilitent la maîtrise de la qualité.

Dès lors, le système d'information équipe les processus internes de l'entreprise au plus près de la pratique professionnelle en articulant, selon une frontière d'ailleurs délicate, les fonctionnalités de l'informatique de communication à celles du traitement des données structurées.

Pour concevoir le traitement des données structurées il avait fallu concentrer l'attention sur les concepts à l'œuvre dans le système d'information et sur les traitements informatiques. Pour concevoir un workflow, il faut concentrer l'attention sur l'enchaînement des tâches des agents et donc sur le processus opérationnel. Celui-ci se complique d'ailleurs avec l'arrivée du multimédia pour les événements externes (utilisation conjointe du courrier, du téléphone, du face-à-face, de l'Internet, de la carte à puce) comme pour les événements internes (*help desk*, Intranet, accès à distance), et aussi avec l'interopérabilité que les partenariats nécessitent entre les systèmes d'information de plusieurs entreprises. La hiérarchie des difficultés invite alors à modéliser en priorité le processus opérationnel : le modèle précisera les concepts et la dynamique sur lesquels se fonde le traitement des données.

4. Le terme « workflow » désigne l'informatisation d'un processus.

Alors qu'auparavant la pratique professionnelle avait été invitée à se construire autour de l'informatique, désormais l'informatique se construit autour de la pratique professionnelle. Ce changement de point de vue s'accompagne, en ce qui concerne l'organisation, de l'émergence dans l'entreprise de spécialistes de la *maîtrise d'ouvrage* du système d'information. Pour prendre en compte de façon exacte le déroulement des processus de production, il faut en effet à la fois une proximité quotidienne avec les agents opérationnels et une rigueur intellectuelle dont le besoin n'avait pas jusqu'alors été ressenti. Ces professionnels de la maîtrise d'ouvrage disposent, pour mettre en forme les processus opérationnels, de langages comme UML⁵, BPML⁶ etc.

De nouveaux problèmes apparaissent alors : comment choisir, si l'on veut un système d'information assez sobre pour pouvoir évoluer, entre les fonctionnalités que l'on fournira et celles sur lesquelles on fera l'impasse ? Comment faire en sorte que les personnels de l'entreprise, ses dirigeants, s'approprient le système d'information, valident ses spécifications, participent à sa définition ? Par ailleurs, si l'informatisation du processus convient aux travaux internes, il sera beaucoup plus difficile d'outiller l'agent du *front office* qui travaille au contact des clients ou des fournisseurs : on ne peut pas prévoir en effet l'ordre dans lequel il devra saisir les données et lancer les traitements. Tout au plus le système d'information pourra-t-il lui fournir une aide contextuelle et la liste des tâches à accomplir, équipée de voyants indiquant pour chaque tâche le degré d'avancement ; le workflow ne débute vraiment qu'au moment où l'agent du *front-office* alimente les événements internes.

* *

Le resserrement des relations entre l'informatique communicante et le traitement des données structurées amène à construire un système d'information « sur mesure », « près du corps », dont la définition et l'évolution adhèrent à la pratique professionnelle. Il permet d'associer aux données leur commentaire, ce qui les rend compréhensibles, facilite leur transformation en information et modifie d'autant leur rôle dans l'entreprise⁷.

Le système d'information assiste alors les diverses personnes de l'entreprise - agents opérationnels, *managers* locaux, concepteurs et stratèges de la direction générale - en fournissant à chacun sur l'entreprise la « vue » qui lui convient : ici les données pour le traitement opérationnel d'un dossier ; là les indicateurs utiles au pilotage opérationnel quotidien ; ailleurs les statistiques qui alimentent les études marketing ou l'analyse stratégique.

Cette évolution rencontre cependant des obstacles. D'une part, comme l'informatique d'une entreprise résulte d'un empilage historique d'applications conçues souvent dans l'urgence, elle est rarement conforme aux exi-

5. « Unified Modeling Language » (Booch, Rumbaugh et Jacobson [19]). Voir page 323.

6. « Business Process Modeling Language ».

7. Le langage XML (eXtensible Markup Language), qui permet d'entrelacer du texte et des données structurées, contribue à cette évolution (Harold et Means [129].)

gences de cohérence du système d'information : il s'en faut de beaucoup que les référentiels et l'administration des données répondent tous aux critères de qualité communément reconnus.

D'autre part, l'histoire a habitué les esprits à une représentation étroite de ce que peut et doit être le rôle de l'informatique. Le choc éprouvé lors de l'arrivée des micro-ordinateurs se renouvelle, sous une autre forme, lorsqu'on met en place la documentation électronique, le multimédia et les workflows : personne ne pensait auparavant que l'informatique pouvait ou devait faire cela et il faut du temps pour que l'on *réalise* (aux deux sens du terme) ces nouvelles possibilités. Enfin, l'évolution de l'informatique confronte l'entreprise à des questions qui touchent à son identité même.

Le tracé des frontières dans l'entreprise, question philosophique

D'après le dictionnaire de Lalande ([108] p. 613), l'une des acceptions du mot « métaphysique » est « connaissance de ce que sont les choses en elles-mêmes par opposition aux apparences qu'elles présentent ». On peut utiliser ce terme pour désigner les idées (pertinentes ou non) concernant la nature de l'entreprise ou celle de l'informatique.

Ces idées influencent la façon dont on trace la frontière entre les activités que l'entreprise doit assurer elle-même et celles qu'elle doit sous-traiter. L'intuition des dirigeants étant déconcertée dans les périodes d'innovation, il peut leur arriver d'adopter des principes antiéconomiques.

Certaines entreprises externalisent ainsi leurs centres d'appel (dont la compétence est alors gaspillée) ou encore la maîtrise d'œuvre de leur informatique (ce qui leur fait perdre la maîtrise de leur plate-forme technique), alors qu'elles conservent l'exploitation des serveurs qu'il serait plus efficace de sous-traiter.

La frontière de l'automatisation est en effet l'objet de convictions très profondes. Certains pensent qu'en équipant les processus opérationnels on dépasse une limite qui n'aurait jamais dû être franchie. Ils éprouvent une horreur instinctive devant le multimédia ou le workflow, horreur qui paraît absurde tant que l'on n'en perçoit pas les raisons. Ainsi s'explique que le même directeur qui lance d'un cœur léger des projets de plusieurs dizaines de millions d'euros refuse un workflow de 100 000 € qu'il considère comme une « usine à gaz ».

Il est vrai qu'il est impossible de tout informatiser : l'informatique doit rester en deçà d'une certaine frontière, mais celle-ci ne passe pas entre le concept (qu'il est légitime d'informatiser) et le processus (qu'il ne faudrait pas informatiser) : le traitement de texte, le tableur, la messagerie et l'Internet, ont montré que l'informatique pouvait se mettre efficacement non seulement au service des tâches de gestion administratives, mais aussi à celui de l'activité quotidienne de l'agent au travail.

Tout système d'information implique une abstraction, un schématisme, donc le renoncement à la finesse sans limites de l'expérience au bénéfice d'une représentation grossière sans doute, mais efficace en pratique. La frontière de l'informatique doit être définie par le degré de détail fonctionnel (et donc conceptuel) qu'il est raisonnable de retenir pour assister l'action des êtres humains, et non par une conception normative (et tissée d'habitudes) du champ légitime de l'informatisation.

Le système d'information a été d'abord consacré à quelques fonctions qu'il remplissait en offrant un luxe de fonctionnalités parfois superflues ; il doit aujourd'hui devenir sobre en fonctionnalités, mais s'étendre jusqu'à fournir aux utilisateurs l'ensemble des fonctions qu'il est utile d'automatiser.

2.2 Entropie du système d'information

On peut distinguer trois phases dans l'histoire de l'informatique : celle des *applications* qui ont automatisé les fonctions administratives dans les années 1950 et 1960 (paie, comptabilité, gestion des stocks etc.) ; celle des *systèmes d'information*, qui démarre dans les années 1970 avec la méthode Merise, mise au point entre 1972 et 1975 ; celle enfin de l'informatisation des *processus* qui recourt volontiers aux langages à objets et généralise dans l'entreprise le « travail assisté par ordinateur », et qui débute dans les années 1980⁸.

Chacune de ces phases peut se représenter par un petit dessin : pour la première, des applications juxtaposées, « en tuyau d'orgue » ; pour la deuxième, qui a ambitionné de corriger le désordre sémantique par la gestion des données de référence, il faut ajouter les bases de données et les référentiels. Un petit diagramme d'activité inspiré d'UML représente convenablement la troisième en évoquant l'informatisation des processus (figure 2.8).

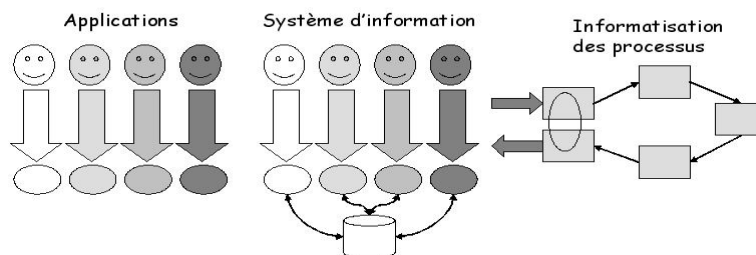


FIG. 2.8 – Les trois phases de l'informatisation

⁸. Bernard Morand, professeur à l'Université de Caen, fournit une utile présentation de l'histoire des méthodes (www.iut3.unicaen.fr/~moranb/).

Dans la troisième phase, la conception d'un système d'information doit obéir à quelques principes élémentaires⁹ : bien définir les domaines d'action de l'entreprise, les processus de production, les « populations » concernées par ces processus, les « classes » à utiliser pour décrire ces populations ; organiser les processus de façon à éviter les doubles saisies, doubles identifications et connexions répétées à des applications diverses ; éliminer les synonymes et homonymes ; construire les référentiels (identifiants, définitions des données) et gérer les données de référence de telle sorte que la sémantique du système d'information soit convenablement maîtrisée...

Mais nos entreprises n'ont pas attendu le système d'information ni le langage UML pour s'informatiser ; les applications conçues « en tuyaux d'orgue » dans les années 1960 et 1970 sont encore là et les refaire coûterait cher¹⁰. Pour corriger leurs défauts les plus criants on leur a associé des référentiels mais ceux-ci ne recouvrent le plus souvent qu'une partie du système d'information (ainsi l'entreprise aura créé un annuaire des personnes et un autre de l'organisation, mais non la nomenclature de ses produits etc.) : d'ailleurs la construction d'un référentiel pose de délicats problèmes de méthode (voir page 265). Les modèles à objets ne concernent aujourd'hui qu'une petite partie des systèmes d'information et doivent être juxtaposées aux réalisations antérieures. Ainsi l'on peut représenter nos systèmes d'information par la figure 2.9 : ils constituent une accumulation géologique de couches obéissant chacune à des priorités et principes différents. Les responsables tentent de se débrouiller pour tirer de cet ensemble disparate la meilleure performance pour le moindre coût.

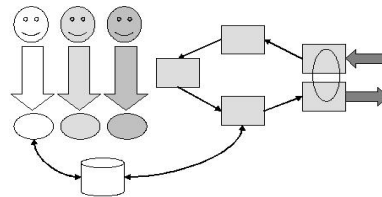


FIG. 2.9 – *Système d'information hybride*

Lorsqu'on parle de système d'information, il faut donc indiquer si l'on parle du système d'information tel qu'il existe *de facto* dans une entreprise, souvent conforme à la figure 2.9, ou du système d'information idéal correspondant au cas hypothétique d'une entreprise nouvelle qui partirait de zéro et qui, de plus, ne commettrait aucune erreur de conception.

Même si nous étions parvenus au terme de l'évolution actuelle, même si les systèmes d'information étaient tous modélisés en UML, le désordre y naîtrait aussi naturellement que l'entropie naît dans la matière.

9. Ces principes sont « élémentaires » au plan de la logique, mais cela ne veut pas dire que leur application soit en pratique aisée : elle est rarement réussie ou complète.

10. Le code source de certaines d'entre elles a été perdu, ou bien on ignore le numéro de la version qui tourne : alors on se garde d'y toucher, espérant que « ça va continuer à marcher ».

Supposons qu'une entreprise ait créé un référentiel de l'organisation et immatriculé ses services, établissements et zones géographiques. Elle a ainsi construit une base de données de référence qui évolue avec les changements de l'organisation. Tout se passera bien si les divers domaines de l'entreprise répliquent ce référentiel sans délai dans leurs processus ou si ceux-ci le consultent lors de chaque utilisation. Cependant les personnes qui écrivent le code seront toujours tentées de construire un référentiel propre à chaque processus : alors le désordre s'installe.

Isabelle Boydens, *Informatique, normes et temps*

Isabelle Boydens [25] a exploré la production et l'interprétation des bases de données selon une approche à la fois technique et historique.

Elle part d'un cas particulier, la base de données de la sécurité sociale belge. Une telle base de données n'est pas un objet « simple », qu'on la considère en termes de qualité, de représentativité, de pertinence ou de lisibilité.

Isabelle Boydens décrit ainsi sa démarche : « Nous avons préalablement sélectionné un ensemble cohérent et représentatif de normes légales dont la base de données assure l'automatisation. Nous avons ensuite analysé et confronté les sources juridiques, les directives et rapports administratifs, les articles de presse, la documentation informatique et enfin le code de programmation correspondant. Nous avons procédé à de nombreuses entrevues avec les gestionnaires et utilisateurs de la base de données. Enfin nous avons longuement observé le processus de gestion et d'interprétation de la base de données opéré dans la pratique et nous y avons nous même participé. Une observation de terrain permet de révéler ce qui n'est ni dit ni écrit, à savoir les mécanismes informels d'interprétation des données. »

Elle a examiné ainsi des aspects essentiels de la pratique des bases de données, aspects très complexes mais que l'on prend rarement la peine d'étudier parce que l'on suppose, bien à tort, qu'une base de données est quelque chose de « simple ».

Les choses se passent en effet souvent ainsi : lors de l'écriture du code, le programmeur y introduit une copie de la table de référence mais, comme il travaille sous une contrainte de délai, il remet à plus tard l'écriture du module qui en assurerait la mise à jour. Puis il oublie ou néglige la nécessité de ce module. Lors de la recette tout se passe bien puisque la table, étant récente, est à jour. Cependant par la suite la table de référence évoluera. La copie ne peut être mise à jour qu'à la main, on oubliera parfois de le faire : alors l'écart se creusera entre les tables et le désordre s'installera.

Supposons ainsi que le système d'information comporte *de facto* plusieurs tables représentant le découpage géographique : le monde a été découpé en « régions » et l'entreprise a donné un nom à chacune d'entre elles.

Le marché évoluant, elle modifie ces régions en faisant passer des pays de l'une à l'autre. Le référentiel est modifié mais les tables utilisées par les divers processus ne sont pas mises à jour simultanément. La définition des régions diffère alors d'un processus à l'autre. Les interfaces signaleront des erreurs, les vérifications et redressements accapareront processeurs et back-offices. L'interprétation des données occasionnera, lors des conversations entre dirigeants, d'interminables perplexités.

Si le désordre concerne plusieurs référentiels (des produits, des clients, des pièces détachées etc.), la pagaïe devient générale. Seule une gendarmerie vigilante (ici une direction de l'architecture ayant l'autorité nécessaire au sein de la DSI) peut maintenir la discipline. Cela rappelle la circulation automobile : le code de la route est connu, mais comme chacun peut être tenté de commettre une faute la peur du gendarme est utile.

« Il n'y a qu'à » mettre des gendarmes pour maintenir l'ordre ? Non, car cela ne résout pas tout : parfois les forces de l'ordre sont débordées. Supposez que le système d'information de votre entreprise soit articulé avec celui d'un partenaire, ce qui suppose l'interopérabilité des deux systèmes d'information : il faudra s'organiser pour faire prendre en compte par ce partenaire les changements de vos référentiels et réciproquement. Le système d'information du partenaire sera peut-être en désordre, mais vos gendarmes n'ont pas le « droit de suite » chez lui : s'il utilise pour classer ses produits une table différente par région, vous serez contraint de les connaître toutes et de suivre leurs errements.

Si l'on ne parvient pas à faire prendre au sérieux ces « histoires informatiques » par les dirigeants, s'ils n'en font pas un des points à négocier lors de l'accord de partenariat, le désordre s'insinuera dans le système d'information par les échanges avec les partenaires.

Une autre source de désordre réside dans les changements de périmètre de l'entreprise. Supposez que votre entreprise en achète une autre. L'alignement des systèmes d'information occasionnera des conflits entre équipes de dirigeants : ce sera à qui prendra le pouvoir, à qui gardera sa place. Pendant cette guérilla il faudra vivre avec un système hétéroclite, des référentiels dont les nomenclatures ne se correspondent pas etc.

Dans l'économie innovante et évolutive d'aujourd'hui les partenariats sont fréquents ainsi que les fusions et absorptions : autant d'occasions pour que l'entropie s'accroisse, quelles que soient la qualité et la bonne volonté des gendarmes.

L'état naturel du système d'information n'est donc pas l'ordre, mais un désordre contre lequel la guerre n'est jamais gagnée. Ce n'est pas une raison pour perdre de vue les principes selon lesquels on doit bâtir et faire évoluer un système d'information, mais il sera en pratique difficile de les respecter exactement.

Comment font les forces de l'ordre lorsqu'elles sont débordées ? Elles louvoient à la recherche du compromis qui permettra le moindre mal : elles pactisent avec une bande pour mettre une autre bande à la raison, elles tolèrent ceci pour pouvoir réprimer cela, elles négocient des appuis auprès de la municipalité, des familles, des associations. Le gendarme se fait diplomate.

Il en est de même du directeur de l'architecture quand les sources de désordre ont un fort débit. S'il parvient un instant à imposer la logique, la discipline, la méthode etc., l'ordre sera de courte durée. Il ne pourra pas se contenter de règles formelles : il doit avoir une sensibilité tactique et « politique » pour limiter la casse et faire en sorte que, quoique désordonné, le système d'information reste assez cohérent pour rendre un service acceptable.

2.3 Limites de l'informatique

En informatique, on dit qu'une opération est « complexe » si elle est logiquement possible mais irréalisable en pratique.

Une **première forme** de cette « complexité » provient de la représentation des nombres dans la mémoire de l'ordinateur. Celle-ci ne pouvant contenir qu'une quantité limitée de chiffres, les calculs informatisés portent sur un sous-ensemble des nombres rationnels, approximation des nombres réels. La précision des calculs est donc limitée même si le nombre de chiffres que contient la mémoire est élevé. Il en résulte de grandes difficultés mathématiques¹¹.

Une **deuxième forme** de « complexité » est liée au nombre de calculs élémentaires que nécessite une opération. En notant n le cardinal de l'ensemble sur lequel on travaille, on dira que la « complexité » est linéaire si elle demande de l'ordre de n calculs, quadratique si elle en demande de l'ordre de n^2 , « exponentielle » si elle en demande de l'ordre de e^n ou, pire, de n^n .

Si par exemple il faut réaliser un calcul élémentaire sur chaque couple d'éléments de l'ensemble, la « complexité » est quadratique car le nombre des couples est égal à $n(n-1)/2$. S'il faut calculer sur chacune des parties de l'ensemble, la « complexité » est exponentielle car le nombre des parties est égal à 2^n . Enfin, si l'on doit faire un calcul sur chacune des permutations des éléments de l'ensemble, dont le nombre est $n!$ (factorielle de n), la « complexité » est de l'ordre de n^n car $n! \sim \sqrt{2\pi n}(n/e)^n$ (formule de Stirling).

Certains problèmes à la formulation très simple peuvent exiger une durée de calcul de l'ordre de l'âge de l'univers : c'est le cas du « problème du commis voyageur » dès que n atteint quelques dizaines (pour trouver l'iti-

11. « Le calcul en virgule flottante est par nature inexact et les programmeurs peuvent facilement s'y tromper : alors la réponse de l'ordinateur sera pratiquement formée de "bruit" à l'état pur. L'un des principaux problèmes de l'analyse numérique est d'évaluer par avance l'exactitude des résultats que fournissent certaines méthodes de calcul. On rencontre ici un problème de crédibilité : nous ignorons jusqu'à quel point nous pouvons faire confiance à l'ordinateur. Les utilisateurs débutants traitent ce problème en considérant implicitement l'ordinateur comme une autorité infaillible : ils ont tendance à croire significatifs tous les chiffres imprimés sur un listing. Les utilisateurs blasés ont l'approche opposée : ils craignent toujours que les résultats n'aient pratiquement aucune signification. Plusieurs bons mathématiciens ont tenté d'analyser rigoureusement une série de calculs en virgule flottante, mais ils y ont rencontré des difficultés tellement immenses qu'ils ont dû se contenter d'arguments de simple vraisemblance. » (Knuth [105] vol. 2 p. 229).

néraire optimal passant par plusieurs villes, il faut calculer la longueur de $n!$ itinéraires)¹².

Cette approche de la « complexité » permet d'évaluer la faisabilité d'un calcul : si n dépasse quelques centaines (c'est le cas de la plupart des bases de données d'une entreprise), un calcul linéaire sera facile, un calcul quadratique difficile et un calcul exponentiel impossible. Le programmeur qualifié arrive parfois à rendre possible un traitement qui, programmé de façon sommaire, aurait été impossible ou difficile : s'il s'agit par exemple de faire un tri, un calcul rustique sera quadratique mais un calcul bien conçu sera d'ordre $n \log(n)$ seulement - et pour traiter le problème du commis voyageur on peut également utiliser des algorithmes qui économisent du calcul.

Ces deux formes de « complexité » informatique n'ont rien à voir avec la complexité du réel qu'aucun système fini ne peut décrire : en effet, même si l'opération qui consiste à répéter un grand nombre de fois un calcul élémentaire demande un temps très long, elle n'est pas au plan logique plus complexe que le calcul élémentaire lui-même, et celui-ci est aussi simple que l'idée qui a guidé sa conception. Il en est de même pour les difficultés mathématiques liées à la représentation des nombres réels par des nombres rationnels. La « complexité » informatique est un homonyme de la complexité du réel, c'est pourquoi nous avons utilisé les guillemets pour la noter.

Par contre la **troisième forme** de « complexité » provient des limites de la logique elle-même : comme l'a démontré Gödel, aucun système logique ne peut contenir la démonstration de toutes les vérités. Le domaine de la pensée pure est donc lui aussi complexe, puisque aucun système fini ne peut en rendre compte. On peut déduire du théorème de Gödel un des résultats les plus importants de la théorie de l'informatique (Sipser [189] p. 165) : *il est impossible de concevoir le programme qui serait capable de vérifier tous les programmes*. Supposons en effet qu'un tel programme P existe.

- 1) Si le programme A est juste, $P(A) = v$ (v pour « vrai »).
- 2) Soumettons à P le programme $G = [P(G) = f]$ (f pour « faux »).
- 3) Si $P(G) = v$, le programme $P(G) = f$ est faux ; donc $P[P(G) = f] = P(G) = f$, ce qui est contraire à l'hypothèse.
- 4) Si $P(G) = f$, alors $P[P(G) = f] = P(G) = v$, ce qui est encore contraire à l'hypothèse.
- 5) Ainsi G ne peut pas être vérifié par P : il ne peut donc pas exister de programme capable de vérifier tous les programmes.

12. Si l'ordinateur fait mille calculs par seconde il faudrait une durée égale à l'âge de l'univers (12 milliards d'années) pour trouver, en considérant tous les itinéraires possibles, le meilleur itinéraire entre 22 villes. Si l'ordinateur fait mille milliards (10^{12}) de calculs par seconde, il faudrait cette même durée pour trouver le meilleur itinéraire entre 28 villes.

Théorème de Gödel

Kurt Gödel [63] a démontré que si l'on construit un système logique pour formaliser la théorie des nombres entiers, ce système contient au moins une formule A telle que ni A , ni sa négation $\text{non-}A$ ne peuvent être démontrées dans le cadre de ce système.

Russell et Whitehead [217] avaient tenté de fonder l'ensemble de la logique sur une base axiomatique. Le théorème de Gödel prouve que ce but est hors d'atteinte : quel que soit le nombre (fini) des axiomes retenus pour fonder un système logique, il existera toujours des propositions vraies qu'il est impossible de démontrer dans le cadre de ce système.

La démonstration de Gödel est très technique. Voici une description schématique de son raisonnement, tel qu'il le présente lui-même dans l'introduction de son article :

- 1) Supposons qu'il existe une Théorie Complète (TC) fondée sur un nombre fini d'axiomes et permettant, si l'on considère une phrase quelconque, de décider sans jamais se tromper si cette phrase est vraie ou non.
- 2) Considérons la phrase « TC ne dira jamais que la présente phrase est vraie ». Nommons cette phrase G , ce que nous noterons : $G =$ « TC ne dira jamais que G est vraie ».
- 3) Soumettons G à TC et demandons à TC de dire si G est vraie ou non.
- 4) Si TC dit que G est vraie, alors G est fausse. Mais alors comme TC a dit que G était vraie, TC a commis une erreur. Cependant par hypothèse TC ne se trompe jamais. Donc TC ne dira jamais que G est vraie.
- 5) Si « TC ne dit jamais que G est vraie », G est vraie. Mais d'après ce que nous venons de voir TC ne pourra jamais le dire.
- 6) Il ne peut donc pas exister de Théorie Complète, c'est-à-dire de théorie permettant, quelle que soit la phrase que l'on considère, de dire si elle est vraie ou non.

Ce raisonnement rappelle le paradoxe fameux mettant en scène un Crétois qui dit « les Crétois ne disent jamais que des mensonges ». Ce paradoxe montre que tout système formel auto-applicable (c'est-à-dire capable d'énoncer des formules sur lui-même) est soit incomplet, soit inconsistant.

Pour prouver que l'arithmétique est auto-applicable, Gödel a mis au point un procédé permettant d'associer de façon bi-univoque un nombre entier à chaque proposition de l'arithmétique, puis il a démontré qu'il existait un entier n qui était le codage de la proposition « la formule n est indémontrable ».

Certes on peut définir des méthodes de vérification efficaces et les outiller de telle sorte qu'elles soient faciles à utiliser. Mais ces méthodes, aussi efficaces soient-elles, ne garantissent pas que tous les programmes qu'elles valident soient corrects : « Le test d'un programme peut servir à montrer la présence des bogues, mais jamais pour montrer qu'il n'y en a aucune »¹³.

Michael Sipser, *Introduction to the Theory of Computation*

Ce livre [189] présente la théorie de l'informatique : la machine de Turing, la théorie des langages, les algorithmes, la décidabilité, le fait qu'il n'existe pas de programme qui sache vérifier les programmes, le chiffrement à clé publique, les ordres de grandeur qui distinguent le faisable (polynomial) de l'impossible (exponentiel) etc.

Définitions, lemmes et théorèmes s'enchaînent. Sipser est un grand pédagogue. Sans renoncer à la rigueur, il conduit son lecteur le long d'un chemin facile, vérifiable ; en partant d'automates simples il fait découvrir les langages, les grammaires, la machine de Turing, la décidabilité, la récursion, la complexité dans le temps (durée des calculs) et dans l'espace (taille de la mémoire nécessaire) etc.

Sipser écrit deux fois la même démonstration : une fois en langage courant pour donner une idée générale (ce qu'il appelle *proof ideas*) ; la deuxième fois en notation formelle. Lorsqu'on lit un chapitre la première fois, on se contente des *proof ideas* pour avoir une idée d'ensemble. Puis on lit les démonstrations en entier pour se familiariser avec les notations.

2.4 Quelle intelligence ?

« These machines have no common sense; they do exactly as they are told, no more and no less. This fact is the hardest concept to grasp when one first tries to use a computer. » (Donald E. Knuth, *The Art of Computer Programming* [105] volume 1, p. v)

« (...) es realmente el chimpancé que quiere aprender a leer, un pobre tipo que se da con la cara contra las paredes, y no se convence, y vuelve a empezar. Ah, pero si un día el chimpancé se pone a leer, qué quiebra en masa, qué desparramo, qué sálvese el que pueda, yo el primero¹⁴. » (Julio Cortázar, *El perseguidor*, Alianza Editorial 1993, p. 63).

13. Edsger W. Dijkstra, « Notes on structured programming » [44].

14. C'est comme si un chimpanzé voulait apprendre à lire : le pauvre type se casse le nez contre des murs, se refuse à admettre l'évidence et recommence. Mais si un jour ce chimpanzé arrive à lire, quelle panique, quelle débâcle, quel sauve-qui-peut, moi le premier !

Parmi les fausses questions que conduit à poser un concept mal bâti se trouve celle de l'intelligence des ordinateurs. Elle suscite des émotions extrêmes et opposées. « Où est l'homme là-dedans ? » demande avec angoisse l'individualiste qui se pose en humaniste (Virilio [209]). D'autres, tout aussi émotifs mais misanthropes, se félicitent de voir la machine éliminer l'être humain qu'ils jugent peu fiable et moralement douteux (Truong [205]). D'autres enfin croient devoir utiliser un vocabulaire emphatique pour célébrer les nouveautés qu'apporte l'ordinateur (Lévy [121]). D'honorables philosophes¹⁵ et d'excellents ingénieurs¹⁶ disent ainsi des choses qui ne rendent pas un compte exact du phénomène de l'informatisation.

Pour éclairer la question il sera utile de se référer de nouveau à la culture américaine. Nous considérerons ici non plus l'entreprise, mais l'armée (Desportes [47] p. 155) : tout comme l'entreprise l'armée est consacrée à l'action, mais sous sa forme la plus urgente et la plus dangereuse. C'est donc dans les doctrines militaires que l'on rencontrera l'expression la plus claire de la doctrine d'action d'une nation, doctrine qui s'appliquera ensuite dans tous les domaines et en premier dans l'entreprise, lieu géométrique de l'action organisée.

La doctrine militaire américaine s'appuie sur le *Précis de l'art de la guerre* [92] du Suisse Antoine Jomini (1779-1869) et, à un moindre degré, sur le *Vom Kriege* [37] du Prussien Carl von Clausewitz (1780-1831). Ces deux généraux avaient observé la stratégie de Napoléon, étudié celle de Frédéric II, et en avaient tiré les leçons. Rivaux sur le terrain de la théorie, ils se sont beaucoup emprunté l'un à l'autre. Leurs théories sont donc plus proches que les interprétations qui en sont données : mais, quand il s'agit d'évaluer une influence, l'interprétation d'une théorie a plus de poids que son contenu même. On peut ainsi associer les noms de Jomini et Clausewitz à deux modèles contrastés.

Selon le modèle de Jomini, tout problème doit être ramené à ses éléments rationnels puis traité de façon scientifique : Jomini entend maîtriser l'art de la guerre en résumant ce qu'elle a de complexe par quelques principes et axiomes évidents. Cette approche satisfait le pragmatisme des Américains : culte de l'offensive, concentration des masses au point décisif pour anéantir l'adversaire, professionnalisation d'une stratégie qui a pour but de gagner les batailles et non de servir par les armes un projet politique. La guerre est un *job* que la nation confie au militaire et que celui-ci exécute.

L'école qui se réclame de Clausewitz ne répudie pas cette modélisation mais postule une continuité entre la guerre et la politique, idée difficilement assimilable par le pragmatisme américain. Cette école insiste par ailleurs sur les conditions de l'action, ces « incertitudes », ces « frictions », ce « brouillard » que Jomini suppose négligeables mais qui en pratique né-

15. « Aujourd'hui notre mémoire est dans le disque dur. De même, grâce au logiciel, nous n'avons plus besoin de savoir calculer ou imaginer. L'humain a la faculté de déposer les fonctions de son corps dans les objets. Et il en profite pour faire autre chose. » (Michel Serres dans *L'Expansion*, 20 juillet 2000).

16. « Dans 10 à 20 ans, l'ensemble des ordinateurs interconnectés aura des capacités intellectuelles qui dépasseront l'homme dans tous les domaines. » (Jean-Paul Figer).

cessitent, pour éclairer la décision, un travail permanent d'interprétation et de synthèse.

Le modèle de Jomini, rationnel et planificateur, incite à l'automatisation parfaite : les êtres humains exécutent les ordres de l'automate pré-programmé qui, sur la base d'une information claire et complète, établit de façon optimale le programme de leur action. À la limite, le stratège n'est rien d'autre que le porte-voix de l'ordinateur. Mais le modèle de Clausewitz, plus complexe, invite au contraire à articuler l'automate et l'être humain car seul ce dernier, s'il a été bien formé, pourra dans une situation imprévue interpréter des rapports incomplets ou fallacieux et prendre la décision juste.

La discussion sur l'intelligence de l'ordinateur s'éclaire si on la situe sur cette toile de fond doctrinale. Soit on suppose les hypothèses du modèle de Jomini vérifiées (information parfaite, planification parfaite etc.) et alors l'action peut être préparée à loisir par un programme et, dans le feu de l'urgence, déterminée au mieux par l'automate qui calcule vite et sans erreurs. Soit on suppose au contraire que la situation comporte une part d'incertitude, de « brouillard », et alors la contribution du cerveau humain est nécessaire.

De ces deux modèles, lequel est le bon ? Cela dépend du cas que l'on considère. S'il s'agit d'assurer la maintenance d'un équipement, le modèle de Jomini semble s'imposer, sous la seule réserve que le technicien émette une alarme en cas d'imprévu. Mais si l'ordinateur permettait de supprimer l'incertitude, cela se saurait dans les salles de marché : les informaticiens auraient fait fortune à la Bourse et du coup la Bourse aurait cessé d'exister¹⁷.

Si nous reprenons la distinction proposée page 67 entre « événement interne » et « événement externe », nous pourrions dire que le modèle de Jomini correspond bien aux événements internes, à la conduite du processus de production de l'entreprise, alors que le modèle de Clausewitz s'applique aux événements externes, que l'entreprise ne peut ni maîtriser ni organiser entièrement : relations avec les clients et avec les fournisseurs, conditions de la concurrence.

Sur ces événements externes, et même si l'information sur le passé et le présent était parfaite - ce qui n'est jamais le cas - l'anticipation comporterait encore une part d'incertitude. Les conditions pratiques de l'action stratégique, celle qui porte non sur l'exécution des tâches mais sur leur conception, sur le positionnement de l'entreprise, sont donc plutôt celles que décrit le modèle de Clausewitz. Le modèle de Jomini conforte la confiance en soi du professionnel mais si celui-ci l'applique aux événements externes il prend le risque de se trouver désarmé devant l'imprévu.

* *

Ceux qui conçoivent des ordinateurs savent faire la différence entre l'ordinateur et l'être humain. John von Neumann¹⁸ était arrivé à l'informatique

17. De même, les casinos auraient fermé si le déterminisme ne comportait pas d'exception, car les physiciens y auraient fait fortune.

18. von Neumann [147]. *The Computer and the Brain* est le dernier ouvrage de von

par la physique et se servait de l'ordinateur pour résoudre par simulation des problèmes trop complexes pour être traités analytiquement. C'est lui qui, avec Church et Turing, a conçu les ordinateurs dont nous nous servons aujourd'hui. Certaines de ses réflexions sont à l'origine de l'intelligence artificielle.

Le cerveau lui a fourni une image de l'ordinateur : tout comme le cerveau, l'ordinateur est doté d'une mémoire, d'une capacité de traitement et d'organes sensoriels (entrées et sorties). Cependant le cerveau, être biologique, progresse à l'échelle de la dizaine de milliers d'années, alors que l'ordinateur, être artificiel, progresse à l'échelle de la dizaine d'années, donc beaucoup plus vite. Certains en ont conclu que l'ordinateur, « cerveau électronique », avait vocation à supplanter le cerveau humain.

« Comment pouvez-vous affirmer, disent-ils, que l'ordinateur sera à jamais incapable d'avoir des émotions et de se comporter comme un être humain ? »

Du point de vue poétique toute parole qui éveille l'intuition est légitime car la poésie relève d'une démarche mentale antérieure au raisonnement. Mais certaines phrases sont impropres à alimenter le raisonnement parce qu'elles ne sont pas « falsifiables », pour utiliser le vocabulaire de Popper. « Y a-t-il une vie après la mort ? » La réponse peut être « oui » ou « non » sans que l'expérience puisse trancher et la conviction intime du croyant lui-même oscille entre ces deux réponses. Donc même si cette question est un thème intéressant pour la rêverie, du point de vue de la réflexion elle est futile. « L'intelligence des ordinateurs atteindra-t-elle, dépassera-t-elle celle des êtres humains ? » est une phrase du même type. On peut y répondre par oui, par non ou osciller entre les deux réponses sans pouvoir trancher par l'expérience puisque celle-ci se situe dans un futur indéfini et ne peut s'appuyer sur aucun précédent.

Il existe entre l'ordinateur et l'être humain une différence plus grande qu'entre l'être humain et les primates qui lui sont génétiquement les plus proches. Se pourrait-il ainsi qu'un chimpanzé apprenne un jour à parler ou à lire (cf. la citation de Cortázar page 86) ? Une telle hypothèse est tout aussi plausible que celle concernant l'intelligence de l'ordinateur.

* *

Hypothèse pour hypothèse, on est libre de choisir. Ne serait-il pas alors plus fécond de postuler qu'il existe entre l'ordinateur et le cerveau humain une différence de nature, une différence infranchissable ?

Cette hypothèse a plusieurs avantages :

- elle est conforme à l'expérience présente, car aucun des ordinateurs existants ne se comporte comme un être humain ;
- elle indique une piste à l'action : si l'ordinateur et l'être humain sont différents, il importe de les articuler ;
- enfin, elle fournit à la pensée un cadre utile pour préparer cette action :

Neumann. Miné par le cancer dont il est mort en 1957, il n'a pas pu l'achever. Il lui accordait beaucoup d'importance.

il est possible de penser l'articulation de deux êtres différents alors qu'il serait impossible de penser l'articulation de deux êtres que l'on suppose identiques, car la pensée les confond.

Von Neumann a perçu les limites de l'analogie entre l'ordinateur et le cerveau humain. Le mécanisme du cerveau, dit-il, est plus lent et davantage sujet à l'erreur que celui de l'ordinateur ; mais la structure du cerveau est, elle, plus riche que celle de l'ordinateur. Le cerveau met en œuvre des processus parallèles et analyse les signaux de façon statistique, notamment en évaluant des corrélations. Cela lui confère une grande rapidité et une grande fiabilité malgré les défauts que comporte son mécanisme. En outre la mémoire du cerveau, qui utilise une grande diversité de supports, semble pratiquement infinie même si la mémoire consciente est, à chaque instant, limitée.

Ainsi notre héritage génétique nous aurait doté d'un automate naturel dont les capacités diffèrent essentiellement de celles d'un ordinateur. Notre cerveau est supérieur à l'ordinateur pour certaines fonctions (interpréter, synthétiser, comprendre, expliquer, décider, concevoir), inférieur et même très inférieur pour d'autres (classer et trouver une information, calculer, recopier, transcoder).

La question pratique à laquelle nous sommes confrontés n'est pas de mettre au point un ordinateur dont le comportement ne différerait en rien de celui d'un être humain - but peut-être hors de portée et en tout cas éloigné des priorités actuelles - mais de réaliser efficacement l'articulation entre l'« être humain organisé » (car bien sûr il ne suffit pas de considérer un seul cerveau : il faut considérer la société entière avec ses institutions et, parmi elles, l'entreprise) et l'« automate programmable doué d'ubiquité », forme que l'ordinateur a prise grâce au réseau.

2.4.1 Turing a-t-il perdu son pari?

Dans un article qui a eu une immense influence (Turing [206]) Alan Turing a soutenu qu'il était possible de concevoir une expérience montrant que l'intelligence de l'ordinateur ne pouvait pas être distinguée de celle d'un être humain. Le *pari de Turing* a éveillé l'ambition de l'intelligence artificielle.

Pour répondre à la question « est-ce que les machines peuvent penser ? » il faut bien sûr pouvoir faire abstraction de l'apparence physique. Turing propose donc un « jeu de l'imitation » qu'il définit ainsi¹⁹ : « Le jeu de

19. « [The imitation game] is played with three people, a man (*A*), a woman (*B*), and an interrogator (*C*) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels *X* and *Y*, and at the end of the game he says either "*X* is *A* and *Y* is *B*" or "*X* is *B* and *Y* is *A*". The interrogator is allowed to put questions to *A* and *B* (...) We now ask the question, "What will happen when a machine takes the part of *A* in this game?" Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, "Can machines think?" (...) I believe that in about fifty years' time it will be possible

l'imitation se joue à trois personnes, un homme (A), une femme (B) et un interrogateur (C) qui peut être de l'un ou l'autre des deux sexes. L'interrogateur se trouve dans une pièce séparée des deux autres. Le but du jeu est pour l'interrogateur de deviner lequel de ses deux interlocuteurs est un homme et lequel est une femme. Il les désigne par les matricules X et Y ; à l'issue du jeu il dit soit " X est A et Y est B ", soit " X est B et Y est A ". Il peut poser des questions à A et B (...) Maintenant nous nous demandons "Que peut-il arriver si l'on fait tenir par une machine le rôle de A dans ce jeu?" Est-ce que l'interrogateur se trompera aussi souvent que lorsque la partie se joue entre un homme et une femme? Ces questions remplacent notre question initiale, "Des machines peuvent-elles penser?" (...) Je crois que dans cinquante ans environ il sera possible de programmer des ordinateurs ayant une mémoire de l'ordre de 10^9 de telle sorte qu'ils jouent tellement bien au jeu de l'imitation qu'un interrogateur moyen n'aura pas plus de 70 chances sur cent de les identifier de façon exacte après les avoir questionnés pendant cinq minutes. (...) La seule façon satisfaisante de prouver cela, c'est d'attendre la fin du siècle et de faire l'expérience que je viens de décrire. »

Pour comprendre la nature du test de Turing, il faut réfléchir un instant à son énoncé. Si la différence entre A et B est évidente, l'interrogateur ne se trompera jamais : la probabilité qu'il ne fasse pas d'erreur est donc égale à 1. Si la différence entre A et B est insensible, l'interrogateur se trompera une fois sur deux (il faut supposer qu'en cas de doute il tire sa réponse à pile ou face) : la probabilité qu'il ne fasse pas d'erreur est alors égale à 0,5.

Le test peut donc être caractérisé par la fréquence des résultats exacts, qui appartient à l'intervalle $[0,5, 1]$ (figure 2.10). Turing ne dit pas que le test sera réussi si cette fréquence est de 0,5, valeur qui correspond au cas où l'on ne pourrait absolument pas distinguer l'ordinateur de l'être humain : il dit que le test sera réussi si cette fréquence est comprise dans l'intervalle $[0,5, 0,7]$, c'est-à-dire *si l'interrogateur a confondu l'ordinateur avec un être humain dans au moins 30 % des cas*.

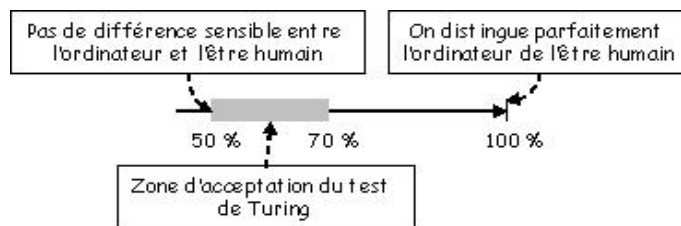


FIG. 2.10 – Échelle du test de Turing

to programme computers, with a storage capacity of about 10^9 , to make them play the imitation game so well that an average interrogator will not have more than 70 per cent chance of making the right identification after five minutes of questioning. (...) The only really satisfactory support that can be given for [this] view (...) will be that provided by waiting for the end of the century and then doing the experiment described. »

Ce test est peu exigeant : il ne dure pas plus de cinq minutes et le seuil d'efficacité est modeste. Il est donc audacieux de prétendre qu'une telle expérience, si elle réussissait, autoriserait à affirmer que des machines puissent penser.

Quoi qu'il en soit, Turing a formulé à la fois un pari et le test qui permet de le vérifier. Nous pouvons faire le test, puisque la fin du siècle est passée. Certes l'ordinateur s'est révélé redoutable au jeu d'échecs ; mais dans un travail aussi « simple » que la traduction d'un texte littéraire il fournit un résultat tellement médiocre que l'examineur moyen le distinguera immédiatement d'un traducteur humain. Si nous prenons Turing au mot, nous pouvons donc dire qu'il a perdu son pari. Mais ce serait là une réfutation peu satisfaisante : quelqu'un d'autre pourrait prendre la relève et parier de nouveau en fixant comme échéance la fin du XXI^e siècle ou celle du troisième millénaire, ce qui laisserait la question en suspens pendant un long délai.

La meilleure réfutation de Turing s'appuie sur la définition de l'ordinateur qu'il donne lui-même :

« On peut expliquer l'idée qui se trouve derrière les ordinateurs en disant qu'ils sont conçus pour réaliser toutes les opérations qui pourraient être faites par un calculateur humain. Le calculateur humain est supposé suivre des règles fixes ; il n'a pas le droit de s'en écarter le moins du monde. Nous pouvons supposer que ces règles lui sont fournies dans un livre qui sera modifié chaque fois qu'on veut lui faire faire un nouveau travail. Il dispose pour faire ses calculs d'une quantité illimitée de papier. Il peut aussi faire ses additions et multiplications sur une machine à calculer, mais c'est sans importance. Le livre de règles qu'utilise notre calculateur humain est évidemment une fiction commode : en fait, les vrais calculateurs humains se rappellent ce qu'ils ont à faire. Si on veut faire imiter par une machine le comportement d'un calculateur humain, on n'a qu'à demander à celui-ci comment il fait, puis traduire sa réponse en une table d'instructions. Construire une table d'instruction, c'est ce que l'on appelle "programmer". "Programmer une machine pour réaliser l'opération A" équivaut à introduire dans la machine la table d'instructions qui convient pour qu'elle réalise A²⁰. »

Si Turing définit ainsi l'ordinateur, puis affirme par ailleurs qu'il sera difficile de distinguer au « jeu de l'imitation » un ordinateur d'un être humain, cela implique qu'il sera tout aussi difficile de distinguer un calculateur

20. « The idea behind digital computers may be explained by saying that these machines are intended to carry out any operations which could be done by a human computer. The human computer is supposed to be following fixed rules; he has no authority to deviate from them in any detail. We may suppose that these rules are supplied in a book, which is altered whenever he is put on to a new job. He has also an unlimited supply of paper on which he does his calculations. He may also do his multiplications and additions on a "desk machine," but this is not important. The book of rules which we have described our human computer as using is of course a convenient fiction. Actual human computers really remember what they have got to do. If one wants to make a machine mimic the behavior of the human computer in some complex operation one has to ask him how it is done, and then translate the answer into the form of an instruction table. Constructing instruction tables is usually described as "programming". To "programme a machine to carry out the operation A" means to put the appropriate instruction table into the machine so that it will do A. »

humain, assujetti au respect de règles fixes consignées dans un cahier, d'un être humain ordinaire qui, n'étant pas assujetti à de telles règles, est libre de prendre des initiatives et de s'adapter à l'imprévu.

Il s'agit donc finalement de savoir si l'on suppose l'imprévu négligeable ou non : on retrouve ici le débat entre les écoles de pensée qui s'inspirent l'une de Jomini, l'autre de Clausewitz. Il est possible de rédiger le cahier de règles de telle sorte que le calculateur dispose de consignes lui permettant de faire face à une grande diversité de situations (c'est ainsi que l'ordinateur joue aux échecs), mais la liste des situations que décrit le programme est nécessairement *finie* alors que l'imprévu potentiel que comporte le rapport d'un être humain avec le monde de la nature est *illimité* : l'être humain « se débrouille » lorsqu'il rencontre un imprévu, là même où l'ordinateur (ou son équivalent le calculateur humain, pur exécutant) ne disposerait pas de règles et ne saurait donc que faire : c'est ce type de difficulté que l'ordinateur rencontre en traduction automatique.

C'est le sens de la réfutation de Turing par Popper²¹ (Popper [163]) :

« Turing a affirmé qu'il était impossible, par principe, de distinguer les hommes des ordinateurs à partir de leurs activités observables (leurs comportements), et il a défié ses adversaires de spécifier une activité ou un comportement humain observable quelconque qu'un ordinateur serait, par principe, incapable de réaliser. Mais ce défi est un piège intellectuel : spécifier un type de comportement reviendrait à établir une spécification pour la construction d'un ordinateur. En outre, si nous utilisons et construisons des ordinateurs, c'est parce qu'ils sont capables de faire beaucoup de choses que nous ne savons pas faire, tout comme je me sers d'un crayon ou d'un stylo pour calculer une addition que je suis incapable de faire de tête. "Mon stylo est plus intelligent que moi", disait souvent Einstein. Mais ceci ne prouve pas qu'il soit impossible de le distinguer de son stylo. »

Turing fut un génie et aussi un être humain dont les souffrances suscitent la compassion²². Il a contribué à la naissance de l'informatique, bouleversement dont nous n'avons pas fini d'explorer les conséquences. Il est compréhensible que le témoin d'un événement d'une telle importance ait eu du mal à en circonscrire la portée. Constatant la rapide croissance des possibilités de l'ordinateur, Turing a extrapolé, *et il s'est trompé*.

* *

Les Grecs avaient découvert la puissance de l'abstraction ; on voit dans *L'Anabase* de Xénophon l'ivresse intellectuelle que cette découverte a provoquée. Platon a été jusqu'à affirmer que seules les idées étaient réelles : l'idée de cheval serait ainsi plus réelle que le cheval qui gambade dans le pré, tout comme l'idée de cercle est plus réelle que le cercle que l'on trace

21. Popper, comme d'autres commentateurs, pousse à l'extrême la formulation du test de Turing qui est comme nous l'avons vu modeste. Mais ce n'est que justice puisque Turing a prétendu que la réussite de ce test permettrait d'affirmer que les machines « pensent ».

22. Homosexuel, Alan Turing (1912-1954) fut comme Oscar Wilde (1854-1900) une victime du système judiciaire britannique. Il s'est très vraisemblablement suicidé (Hodges [84]).

avec un compas. L'idéalisme platonicien a inspiré la philosophie de Plotin et de saint Augustin, la théologie catholique, il a influencé les hommes de la Renaissance et bien d'autres : ce n'est pas parce qu'une représentation est inexacte ou incomplète qu'elle n'a pas d'audience, surtout quand elle est formulée par une personne de grand talent.

Le pari de Turing a eu des effets contrastés. Le parallèle entre le cerveau électronique et le cerveau humain a suscité des recherches fécondes même si les travaux en intelligence artificielle ont donné des résultats décevants. Mais en focalisant l'attention sur l'intelligence de l'ordinateur et donc sur la *similitude* entre celui-ci et l'être humain, ce pari a eu l'inconvénient d'inciter à ne pas se soucier de leurs *différences*.

Quelles sont nos priorités *aujourd'hui* ? Est-ce de rêver à l'intelligence que la machine pourrait, par hypothèse, déployer dans quelques siècles ou quelques millénaires, ou bien d'agir pour articuler au mieux les ressources qu'offre aujourd'hui l'automate programmable (l'ordinateur) avec ce que sait faire l'être humain organisé dans l'entreprise ? Choisir la seconde réponse, c'est dire que le pari de Turing n'était pas pertinent.

* *

Les questions que nous venons d'évoquer n'ont rien de fondamentalement nouveau. Le cerveau humain s'est déjà accommodé de plusieurs « successeurs » potentiels : le langage, qui a une existence propre depuis que des hommes se sont mis à parler ; l'écriture, support de la mémoire ; l'impression, support de la diffusion des textes. Des machines remplacent nos jambes (bateau, bicyclette, automobile, avion), des prothèses assistent nos sens (lunettes, appareils acoustiques, téléphones, télévision). L'élevage et l'agriculture pratiquent depuis le néolithique la manipulation génétique par sélection des espèces.

La bionique, l'intelligence artificielle ne font que venir s'ajouter au catalogue des prothèses qui assistent nos activités physiques ou mentales. N'avons nous pas tendance, par défaut de perspective historique, à exagérer leur nouveauté ?

Il existe entre la complexité du logiciel et celle du cerveau d'un être vivant une différence de nature. Aussi compliqué soit-il, le logiciel est de taille finie puisqu'il s'agit d'un *texte*. Mais toute théorie, aussi puissante soit-elle, reste incomplète ; aucun objet du monde de la nature (et le cerveau en est un) ne peut donc être exactement reproduit par un texte. Si un texte poétique semble parfois nous mettre en relation avec le monde lui-même, c'est parce que notre cerveau le complète par un réseau de connotations qui enrichit l'apport des mots et, au prix d'une imprécision que le logiciel ne saurait tolérer, ouvre à notre rêverie une perspective sans limites.

L'intuition de ceux qui vivent dans un univers de science-fiction ou de dessin animé s'affranchit de l'expérience : dans cet univers-là toutes les métamorphoses sont possibles, toutes les chimères peuvent exister et quiconque évoque une impossibilité serait immédiatement démenti. Mais l'expérience distingue le possible de l'impossible et assigne des bornes à l'action. Il existe dans le monde de l'expérience des questions *pertinentes*, c'est-à-dire utiles à

l'action, et d'autres qui ne le sont pas. Si elles nous détournent des questions pertinentes, nous devons qualifier les rêveries sur l'intelligence des ordinateurs d'*impertinentes*.

Nous qui savons tant bien que mal parler, lire, écrire, compter, domestiquer plantes et animaux, fabriquer produits et outils, communiquer, déposer et retrouver notre mémoire collective dans des encyclopédies etc., devons apprendre à tirer parti de l'automate programmable.

Pour voir clair dans les questions de savoir-faire et de savoir-vivre, d'organisation collective et personnelle que cela pose, il importe de percevoir la frontière qui nous sépare de lui, de discerner ce qu'il sait faire de ce que nous savons faire, de sorte que son insertion dans notre action, dans nos processus, puisse être judicieuse. Il faut pour tracer cette frontière un outil conceptuel aussi précis que le scalpel du chirurgien.

Or les rêveries sur l'intelligence de l'ordinateur brouillent cette frontière. On ne peut pas penser la relation entre deux êtres dont on a postulé l'identité, fût-elle asymptotique. L'intelligence de la machine s'actualisant dans un futur indéfini, l'intuition s'évade des contraintes de l'action en tournant le dos à des questions qui aujourd'hui sautent aux yeux : comment assister nos processus opérationnels ; tirer parti de la conjugaison des données et du commentaire ; fonder la solidité des référentiels ; articuler les médias ; faire interopérer les systèmes d'information de diverses entreprises ; assurer la dialectique du système d'information et de la stratégie etc.

Les écrivains de science fiction, les cinéastes, créent un monde imaginaire ; il leur est facile d'y doter les ordinateurs de facultés extraordinaires, comme l'a fait Stanley Kubrick dans *2001 : odyssée de l'espace*. Ils sculptent ainsi un imaginaire fallacieux²³. Ce n'est pas sans conséquences. Certes, la rêverie est innocente quand elle délasse le praticien expert : il n'est pas dupe des illusions qu'elle comporte et il est lié à l'expérience par le ressort de rappel de son activité professionnelle.

Mais parmi les personnes qui décident en matière de système d'information les praticiens experts sont une minorité. La diffusion médiatique des rêveries sur l'intelligence de l'ordinateur risque de placer les autres sur une orbite mentale d'où il leur sera impossible de revenir au sol, et alors leurs décisions seront follement inadéquates.

Ce n'est pas de rêveries impertinentes que nous avons besoin dans ce domaine difficile, mais de démarche scientifique et de méthode expérimentale.

L'ordinateur ne peut pas réaliser des choses que l'être humain apprend à faire lors des premières années de sa vie : il ne comprend pas le langage humain ordinaire, avec ses allusions et ses connotations. Il ne peut pas faire la synthèse d'un ensemble de faits et en tirer la conclusion. Il ne peut pas prendre de décision. Il n'a pas d'imagination. Si l'on a l'impression qu'il sait faire tout cela, c'est que l'on commet l'erreur de dire « l'ordinateur calcule »

23. « Contrairement à ce que l'on voit dans les films, les logiciels qui existent dans le monde réel sont incroyablement primitifs si on les confronte aux exigences de ce que nous appelons le "simple bon sens". Regardez par exemple la stupidité incroyable des moteurs de recherche sur la Toile lorsqu'on leur demande de trouver des contenus ayant un sens pour l'être humain » (Schneier [180]).

quand on se sert de l'ordinateur pour faire un calcul, ou « l'ordinateur décide » quand on l'utilise pour aider la décision.

Ce n'est pas l'ordinateur qui calcule ni qui décide mais l'utilisateur, qui se fait aider par l'ordinateur pour calculer ou décider. Cette aide rend parfois possibles des choses qui, autrement, auraient été impossibles : ainsi le pilote automatique d'un avion permet de maintenir celui-ci pendant toute la durée du vol dans la position qui minimise la consommation de carburant, alors que pour le pilote humain ce serait aussi délicat que de tenir une assiette en équilibre sur une aiguille, performance qu'il est impossible de réaliser des heures durant.

Pour progresser dans l'utilisation de l'ordinateur il faut se rappeler qu'il obéit à la lettre, sans initiative ni interprétation, aux ordres qui lui sont donnés ; il apporte ainsi à l'être humain une aide précieuse, mais bien délimitée. L'expression « assisté par ordinateur », que l'on utilise pour le dessin, la gestion, la conception etc. (DAO, GAO, CAO), a une portée générale : en tout et pour tout, *l'ordinateur nous assiste* ; les utilisations les plus intelligentes de l'informatique sont celles qui cultivent la relation entre l'ordinateur et nous, à partir d'une claire conscience de la différence qui nous sépare de lui.

La traduction automatique

La traduction automatique ne donne de résultat acceptable que lorsqu'il s'agit de textes formalisés, techniques, précis, sans surprises du point de vue de la syntaxe. Par contre la traduction automatique des textes littéraires, rédigés dans le langage courant et riches en connotations, donnent des résultats ridicules.

Pour voir ce qu'il en est, appliquons le logiciel fourni par AltaVista à un paragraphe de la fameuse lettre ouverte de Bill Gates (Gates [62], voir page 111) :

To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?

On obtient ceci :

À moi, la chose la plus critique sur le marché de passe-temps est en ce moment le manque de bons cours, de livres et de logiciel de logiciel lui-même. Sans bon logiciel et propriétaire qui comprend programmant, un ordinateur de passe-temps est gaspillé. Le logiciel de qualité sera-t-il écrit pour le marché de passe-temps?

Il ne s'agit là que d'un exemple et il existe certainement des logiciels de traduction plus puissants (notamment ceux que l'on étalonne par analyse statistique d'un corpus de textes). Toutefois le risque d'un contresens non seulement grossier, mais absurde, restera toujours élevé.

2.4.2 Articuler l'ordinateur et l'être humain

Pour concrétiser le propos, considérons quelques exemples :

1) Une entreprise veut mettre sur son site Web la fonction d'intermédiation assurée jusqu'ici par une « première ligne » proche des clients et très expérimentée : le client déposera sa demande, les fournisseurs leurs offres, l'informatique fera le rapprochement entre les deux.

Mais il faut mettre l'entreprise en garde. Que va-t-il se passer si elle confie l'intermédiation à un automate? Ne va-t-elle pas y perdre en efficacité?

2) Un logiciel de traduction automatique est (a) coûteux à concevoir, (b) fécond en contresens insupportables à la lecture, (c) moins efficace qu'un logiciel de traduction assistée par ordinateur dont la production coûte mille fois moins cher.

Si vous souhaitez que l'ordinateur fasse tout, vous devez réaliser un programme qui saura traiter les cas particuliers ; il sera d'une complexité monstrueuse et, en fait, il ne parviendra pas à traiter *tous* les cas particuliers.

Si vous articulez l'ordinateur et l'être humain, vous pouvez laisser à ce dernier le soin d'interpréter les cas particuliers en utilisant son discernement. L'ordinateur traitera les cas simples, qui constituent la majorité des affaires ; le programme sera plus simple, donc moins coûteux, plus évolutif, mieux maîtrisable.

3) Il se produit des catastrophes lorsque le système d'information est conçu de telle sorte que les personnes de la première ligne n'aient pas la possibilité d'agir, alors qu'ayant expérience et bon sens elles comprennent correctement chaque cas particulier. Il est impossible d'imaginer à l'avance la gamme des incidents qui peuvent survenir en cas d'automatisation, mais il est certain que des incidents se produiront et que l'absurdité peut se déployer sans limites si on ne ménage pas la possibilité d'une « reprise de main » par un être humain.

* *

L'articulation entre l'automate et l'être humain demande des consultations, de la réflexion, bref une *démarche*.

Certaines personnes disent alors : « Vous proposez une démarche, mais ce qu'il nous faut c'est un *produit* ». Les fournisseurs de logiciels portent une part de responsabilité dans cette erreur de perspective. On a vu, sur la couverture du *Monde Informatique* (n° 839 du 4 février 2000) la photo d'un fromager accompagnée de cette légende : « De quoi avez-vous besoin pour transformer votre business en e-business? Découvrez-le page 11 ». Et à la page 11 se trouvait une publicité contenant ces mots : « Il faut un puissant logiciel pour transformer le business en e-business. Ce logiciel existe, IBM l'a fait ».

Or la première question qu'une entreprise doit se poser n'est pas « quel logiciel choisir », mais « que veulent mes clients », puis « quel rôle dois-je jouer » : la check-list ne commence pas par la technique mais par la stratégie. En suggérant qu'il suffit de prendre un logiciel - le sien bien sûr - IBM oriente ses clients vers la voie de l'échec même si son logiciel est excellent (et IBM a

Démarche de mise en place de l'e-business

« Il ne suffit pas pour votre entreprise d'avoir une présence sur la Toile, fût-elle jolie. Il faut d'abord connaître vos clients et savoir ce qu'ils attendent de vous car la Toile, c'est le pouvoir au client : si vous ne répondez pas comme il le souhaite, clic ! il est parti, vous pouvez lui dire adieu.

« Quel positionnement voulez-vous donner à votre entreprise sur la Toile? jusqu'où voulez-vous pousser la différenciation de votre offre? avec quels partenaires voulez-vous vous associer? quelles relations souhaitez-vous avoir avec vos fournisseurs? jusqu'où entendez-vous pousser l'intégration entre vos affaires et celles de vos partenaires, fournisseurs et clients? Il faut ici une ingénierie d'affaire, avec ses dimensions juridique et financière.

« Souhaitez vous conserver la même périphérie, ou pensez vous qu'il faut externaliser certaines de vos activités? Le e-business va de pair avec un e-management : il faut repenser la personnalité, les priorités, les contours de l'entreprise. Il convient que cette réflexion ne soit pas seulement celle du président-directeur général mais qu'elle soit partagée par les dirigeants, les cadres et finalement par toute l'entreprise : cela suppose des consultations, concertations et validations.

« Enfin quand vous saurez ce que vous voulez faire il faudra s'assurer que c'est faisable. Vos limites sont ici celles de votre système d'information. Si celui-ci est constitué d'une accumulation d'applications hétéroclites fondées sur des définitions incohérentes, si les données de référence ne sont pas gérées, s'il n'existe pas de gestion de configuration, bref si vous n'avez pas un système d'information digne de ce nom, vous aurez du mal à jouer la partie de l'e-business. Ce ne sera pas totalement impossible - il n'est pas indispensable de passer par un ERP^a avant de se lancer sur l'Internet - mais ce sera difficile. Le calendrier de mise en œuvre des fonctionnalités de l'e-business sera articulé avec la mise à niveau de votre système d'information.

« Vous pouvez démarrer tout de suite, mais il vous faudra quelques années pour transformer l'entreprise. »

^a « Enterprise Resource Planning »

d'excellents produits). Les entreprises n'ont que trop tendance à croire que tout problème est technique (c'est-à-dire relève étymologiquement du savoir-faire) et donc que toute solution doit être également technique. Mais avant de savoir faire, il faut savoir ce que l'on veut faire, pourquoi et pour qui on veut le faire. « Pourquoi faire » et « vouloir faire » doivent précéder « savoir faire ». Dire cela, ce n'est pas dénigrer la technique, mais au contraire c'est la respecter assez pour ne pas la faire intervenir à contre-temps.

Les systèmes d'information ne sont pas des automates dont on attend qu'ils règlent tous les problèmes, mais des outils destinés à assister des opérateurs humains. La conception du système d'information doit donc considérer non le seul automate, mais le couple formé par l'automate et l'être humain organisé qu'il assiste.

Éloge du semi-désordre

« Automation always looks good on paper. Sometimes you need real people. » (Veronica Stevenson, à propos de l'échec du système automatique de transfert des bagages à l'aéroport de Denver ; citation du jour, *The New York Times*, 27 août 2005)

Si le système d'information est « parfait » l'entreprise peut paradoxalement devenir inefficace. Voici quelques exemples :

1) Il est judicieux d'équiper une centrale nucléaire d'un logiciel moins complet qu'il ne pourrait l'être. En effet, les lacunes du système d'information obligent les opérateurs humains à faire chaque jour des interventions manuelles. Le jour où se produira un incident imprévu, ils sauront comment faire car ils ont l'habitude de traiter les « pépins ».

Si le système d'information était aussi complet que possible, il se produirait un incident tous les trois ans. Les opérateurs perdraient l'habitude de réagir et feraient confiance au système ; le jour où se produirait un incident ils ne sauraient que faire.

2) La conception des avions est l'enjeu d'un conflit entre ingénieurs et pilotes. La qualité des avions étant élevée, les ingénieurs voient dans le « facteur humain » la cause résiduelle des accidents.

Pour l'éliminer ils souhaitent concevoir l'avion « parfait » qui décollerait, volerait et se poserait sans pilote. Mais les pilotes disent qu'il reste des situations où l'on a besoin du cerveau humain pour synthétiser, arbitrer et décider : l'avion doit comporter des automatismes, mais ceux-ci doivent assister le pilote et non le supplanter.

3) Considérons une administration comme les impôts, la sécurité sociale ou l'ANPE. La réglementation évolue souvent, ce qui exige de modifier le système d'information. La modification est simple s'il s'agit de mettre à jour quelques paramètres, complexe s'il faut redéfinir une partie d'un dossier et introduire des traitements nouveaux.

Il faut de trois à six mois pour introduire une modification complexe dans le système d'information. Si celui-ci est de qualité médiocre, il faudra un an pour corriger les bogues provoquées par la modification. Pendant ce délai la réglementation aura encore changé.

Les agents se sont donc habitués à faire une partie de leur travail sur papier ou sur tableur, puis à saisir les données dans le système d'information. Cela comporte des inconvénients (erreurs de calcul ou de saisie, surcharge de travail, inefficacités diverses etc.) mais ce fonctionnement d'ensemble permet à l'administration d'être réactive et de mettre en œuvre sans délai une politique nouvelle.

4) On a pu, dans certaines entreprises, modéliser la pratique professionnelle des agents pour automatiser leur démarche et gagner en rapidité. C'est ainsi que des banques ont conçu des systèmes experts de gestion de trésorerie. Cependant, si le contexte évolue, le système expert ne saura pas, lui, évoluer et il perdra en efficacité alors qu'un opérateur humain aurait adapté ses méthodes de travail et modifié ses « règles de pouce ».

Il faut donc conserver, à côté du système expert qui fera le gros du travail, des opérateurs humains plus lents sans doute, mais dont le savoir pourra être périodiquement réinjecté dans le système expert pour le mettre à jour.

5) La gestion d'un système d'information (ou d'un projet) navigue entre deux extrêmes. Suivre une méthode oblige à consacrer beaucoup de temps à la production de documents qui décrivent le système d'information sans faire nécessairement progresser son adéquation fonctionnelle. On peut aussi pratiquer l'artisanat « à l'ancienne » : dès qu'un métier a besoin de quelque chose, il demande aux informaticiens de le programmer ; il revient à ceux-ci de gérer l'intendance, le métier ne se souciant pas des problèmes techniques du système d'information.

Si l'on tolère la non-formalisation, les maîtrises d'ouvrage risquent de s'y engouffrer ; si on exige une formalisation complète, l'entreprise s'enlisera dans la production de documents en grande partie superflus. Le moyen terme efficace résulte d'un bon sens qui ne peut pas être entièrement formalisé.

* *

Un système d'information totalement désordonné n'est pas un système (la notion de système implique la cohérence) et ne contient d'ailleurs pas d'*information* car il stocke et produit des données qu'il sera impossible de comparer et donc d'interpréter. Le désordre total, c'est la mort du système d'information qui devient un *machin informe*.

La perfection est une autre forme de mort car elle démobilise les opérateurs humains. Le mieux est l'ennemi du bien : il faut admettre une dose de « non qualité » (apparente) pour que la coopération entre l'automate et l'être humain soit le plus efficace possible.

Le laxisme peut prendre deux formes :

- le formalisme « fait sérieux », mais il aboutit soit à l'inefficacité, soit (conséquence moins dommageable) à la frustration du méthodologue qui ne parvient pas à se faire entendre ;

- le fatalisme, que traduisent l'expression « ça finira bien par tomber en marche » et la phrase attribuée au président Queuille (1884-1970) : « il n'existe pas de problème dont une absence persévérante de solution ne finisse par venir à bout ».

Le semi-désordre est à l'opposé du laxisme : celui qui perçoit la façon dont l'automate et l'être humain s'articulent ne surestime pas les apports du formalisme, il ne s'en remet pas non plus au fatalisme. La perception claire du résultat opérationnel à atteindre guide le choix de ses priorités et l'aide à procéder aux simplifications nécessaires.

L'optimum ainsi atteint découle non de l'application persévérante d'une règle simple, mais d'un arbitrage délicat et continu qui relève de l'art du praticien.

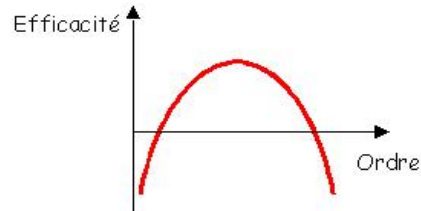


FIG. 2.11 – Relation entre l'ordre et l'efficacité

Une automatisation à 100 % est inefficace, tout comme le désordre absolu ; le maximum d'efficacité est obtenu si l'on sait doser la puissance de l'automate et la souplesse de l'intellect (figure 2.11). Ce maximum est sans doute plat (il existe une plage relativement large de dosages possibles), mais les deux extrêmes sont à éviter.

Il arrive, lorsque l'on dit « semi-désordre » dans une conférence ou un cours, que les auditeurs comprennent « il faut laisser un peu de pagaïe s'installer, ce sera mieux ainsi ». Cela plaît beaucoup à certaines personnes mais c'est un contresens : la recherche de l'articulation efficace entre l'automate et ses utilisateurs requiert une bonne connaissance des conditions pratiques d'exécution du processus et une réflexion approfondie. Elle est aux antipodes de la facilité comme de la pagaïe.

Chapitre 3

Éclairage historique

« Do not be encumbered by history. Go off and do something wonderful. » (Robert Noyce, co-fondateur d’Intel ; source : www.intel.com/intel/)

« Une “révolution technologique” offre-t-elle l’occasion de remettre en cause les situations établies (...) ? Certains dirigeants français l’espéraient dès 1950, lorsque les premiers ordinateurs apparurent en Angleterre et aux États-Unis. Pourtant, les efforts accomplis (...) n’ont abouti qu’à faire de Bull, jadis quatrième constructeur mondial, une firme reléguée au treizième rang et dépendant des technologies américaines et japonaises [alors même que] la France réussissait à s’imposer dans (...) les télécommunications et le logiciel. Expliquer ce paradoxe conduit à analyser à la fois des caractères de longue durée : le purisme de l’école mathématique française, la rareté des compétences, la faiblesse de la recherche industrielle, la priorité des projets militaires sur les logiques de marché limitaient l’aptitude de la France à profiter de la “révolution informatique” (...); et les aspects contingents de cette histoire : échecs locaux et fautes stratégiques ont dépassé les étroites marges d’erreur permises (...) » (Mounier-Kuhn [143]).

L’histoire de l’informatique intéresse un nombre croissant de chercheurs. Le champ ainsi ouvert à la réflexion est immense. Cette histoire ne peut en effet se comprendre que si l’on étudie et relie entre eux ses aspects logique, technique, philosophique, économique, sémantique, sociologique, industriel etc. : il faut décrire à la fois ce qu’elle est, ce que ses promoteurs auraient voulu qu’elle fût, ce que ses utilisateurs en ont fait. Il faudra, pour venir à bout de ce programme, lui appliquer une attention aussi minutieuse que celle que Donald Knuth a consacrée aux algorithmes.

Parmi les chercheurs, on doit citer en France Pierre Mounier-Kuhn. Après avoir soutenu une thèse intitulée *L’informatique en France de la Deuxième guerre mondiale au Plan Calcul*, il anime un séminaire sur l’histoire de l’informatique à l’École pratique des hautes études. Un colloque

international sur l'informatique et les réseaux s'est tenu à Grenoble en novembre 2002¹. En utilisant un moteur de recherche, on trouve sur la Toile de nombreux sites dédiés à l'histoire de l'informatique².

Paul E. Ceruzzi, *A History of Modern Computing*, MIT 1998

Paul Ceruzzi a tenté de construire une vue d'ensemble de l'informatique. Il n'y arrive qu'en partie. D'une part, comme souvent en histoire, il s'arrête au seuil de l'époque actuelle : s'il décrit l'histoire de l'Internet, il ne décrit pas celle des langages à objets, ni des langages de modélisation, ni les outils de *middleware*, ni les changements apportés par l'Intranet dans les entreprises, ni même l'intelligence artificielle qui est pourtant une aventure déjà ancienne.

On souhaiterait par ailleurs qu'il abordât son sujet non seulement sur les plans technique et historique, mais aussi sur le plan philosophique. Comment exprimer autrement que dans le langage de la philosophie des projets visant à construire, relier et utiliser des concepts ? ce qui distingue tel langage de tel autre, ce qui fait la force ou la faiblesse d'outils bureautiques comme le tableur, la messagerie, la documentation électronique etc. ?

La lecture de ce livre est cependant utile. Il apporte des informations jusqu'ici inédites ou difficiles à trouver sur les origines des choix techniques, du vocabulaire, ainsi que sur les mécanismes de la réussite sur ce marché très concurrentiel.

La plupart des travaux érudits sont consacrés aux origines : certains partent des calculs gravés sur les tablettes mésopotamiennes, puis décrivent l'histoire des systèmes de numération (Ifrah [88]), les abaques des calculateurs de l'antiquité, la machine à calculer de Pascal (1623-1662), la machine programmable de Charles Babbage (1791-1871), enfin la machine mécanographique de Herman Hollerith (1860-1929).

Nous n'avons pas la prétention de rivaliser avec ces travaux mais nous nous efforcerons de situer dans l'histoire les questions techniques, sociologiques ou économiques que pose aujourd'hui l'informatique. Cette démarche aide à délimiter ses concepts essentiels et à identifier les ressorts de sa dynamique.

Il ne s'agit pas de collectionner des faits dont le souvenir inspirerait la nostalgie, ni de se repaître de l'image de machines désuètes, mais de prendre assez de hauteur pour s'affranchir du poids du présent³ : l'analyse

1. Entre autres ouvrages, nous devons beaucoup à Philippe Breton [26] Robert Ligonnière [118], Paul E. Ceruzzi [35] et D. Sjobbema [190].

2. Voir en particulier le site de Serge Rossi, histoire.info.online.fr.

3. Les travaux érudits, même quand ils se limitent à un empilage minutieux de documents, témoignages et références, ont pour but d'ouvrir la voie à la réflexion : l'érudition, tout aussi nécessaire que l'expérimentation, prépare comme elle son propre dépassement théorique.

historique, quand elle est poussée jusqu'au passé récent, permet d'élucider le présent et d'éclairer le futur tout en laissant sa part à l'incertitude.

Nous daterons les événements : la chronologie donne un point d'appui au raisonnement en aidant à placer les événements dans leur contexte. Elle aide aussi à identifier les innovations essentielles, celles qui, élargissant le domaine du possible, ont suscité d'autres innovations : certaines dates, qui servent de pivot à l'intelligence des faits, se gravent utilement dans la mémoire.

3.1 Histoire du micro-ordinateur

Nous avons décrit les circonstances de l'invention du microprocesseur chez Intel en 1971 (voir page 47). Un micro-ordinateur, c'est un ordinateur équipé d'un microprocesseur : le micro-ordinateur ne pouvait donc pas naître avant le microprocesseur.

Mais il ne suffisait pas que le micro-ordinateur naquît : il fallait aussi qu'il rencontrât un terrain favorable au succès commercial. Ce terrain a été préparé par les *hackers* des années 1960 et 1970 (Steven Levy [113]), ces pionniers qui ont rêvé le micro-ordinateur et la dissémination de l'informatique longtemps avant qu'ils ne fussent possibles, et qui s'étaient préparés à tirer parti de l'innovation technique dès que celle-ci se produirait. Ils n'étaient pas des pirates, des briseurs de sécurité, des fabricants de virus : le mot « hacker » n'a pris ce sens péjoratif que récemment.

* *

L'informatique était dans les années 1960 l'affaire de professionnels stylés par IBM. Ostensiblement sérieux, ils portaient costume, cravate noire et chemise blanche. Les utilisateurs n'étaient pas autorisés à approcher la machine.

Mais les hackers (que l'on a appelé aussi *hobbyists*) revendiquaient le droit de comprendre comment la machine fonctionne, d'y accéder, de travailler en temps réel et donc de modifier la façon dont on utilisait l'ordinateur. Avant eux, l'ordinateur était une grosse machine sans écran, sans carte sonore, sans traitement de texte, sans tableur, sans réseau etc. Ils ont inventé les procédés qui ont permis d'introduire ces perfectionnements.

Un *good hack*, c'est une astuce qui marche, un procédé qui permet de faire faire quelque chose de nouveau par l'automate - peu importe d'ailleurs si l'on ne sait pas exactement pourquoi ni comment cela fonctionne⁴. La passion des hackers était de créer de *good hacks*, non de les vendre ni de « faire du business ». Ils copiaient sans vergogne les programmes, les modifiaient, les

4. Un hacker remarqua ainsi un jour que lorsqu'il faisait passer un de ses paquets de cartes perforées le lecteur de cartes vibrait en émettant une note de musique. Il repéra les cartes qui causaient ce phénomène et chercha systématiquement les dessins de carte qui pourraient provoquer d'autres notes de musique. Après quoi il programma un petit air et montra à ses amis éberlués qu'un lecteur de cartes pouvait jouer de la musique. Il faudra encore découvrir comment coder le son, comment interfacer un haut-parleur à l'ordinateur etc.

communiquaient. L'*open source* (« logiciel ouvert » ou « logiciel libre »⁵), dont le programme source est fourni et modifiable à volonté, allait pour eux de soi ainsi que la gratuité.

La première équipe de hackers, la plus flamboyante, fut celle du MIT ; puis d'autres équipes de passionnés se sont créées ailleurs : aux Bell Labs d'AT&T, à l'université de Berkeley et, dans les années 1970, au Palo Alto Research Center (PARC) de Xerox. Les hackers du MIT estimaient avoir le droit et même le devoir d'accéder à la machine quelles que soient les interdictions et réglementations, et même s'il fallait fabriquer de fausses clés pour se faufiler la nuit dans un local informatique.

Ils ont mis au point des langages et des méthodes, inventé l'intelligence artificielle etc. Le système d'exploitation Unix (1969), le langage C (1972) mis au point aux Bell Labs ont été d'abord distribués gratuitement ou pour un paiement symbolique, ce qui a permis de les perfectionner rapidement. La souris a été inventée par Douglas Engelbart en 1968, les fenêtres et menus déroulants par Dan Ingals au PARC en 1974. Enfin les hackers ont cherché à mettre l'ordinateur à la disposition de tout le monde d'abord en disséminant des terminaux, puis en mettant au point le micro-ordinateur.

Steven Levy a décrit les rêves et les ambitions de ces passionnés qui travaillaient parfois trente heures d'affilée et sacrifiaient leur santé et leur vie affective à l'exploration des possibilités de l'informatique, au développement d'outils qui permettraient de les concrétiser. Leur imprégnation par le langage informatique les coupait des modes de communication naturels et les isolait des autres êtres humains (voir page 31 les difficultés rencontrées dans la communication entre un hacker et son épouse). Ils avaient une morale, « l'éthique des hackers » (voir page 107). L'émergence du commerce des logiciels au milieu des années 1970 a mis un terme à cette époque qui se prolonge cependant, et ressuscite peut-être, dans l'école du logiciel libre.

Les hackers ont travaillé sur ce qui était à chaque époque le front de taille de l'informatique, sa « couche critique ». Il s'est agi dans les années 1960 de mettre au point les langages qui permettraient de diversifier les utilisations de l'ordinateur : la couche critique était alors celle du *logiciel*. Dans les années 1970 il s'est agi de mettre l'ordinateur dans les mains de chacun en s'appuyant sur les tout nouveaux micro-processeurs : la couche critique fut alors celle du *matériel* ; une fois celle-ci traitée, il a été possible de réaliser les développements qui ont conduit au micro-ordinateur en réseau d'aujourd'hui, avec ses interfaces graphiques, son équipement multimédia et l'Internet.

La couche critique est aujourd'hui celle de *l'utilisation* collective, organisée, de l'informatique par les entreprises, la société et les individus. L'utilisation individuelle pose des questions qui relèvent de la psychologie ; l'utilisation collective pose des questions qui relèvent de la sociologie et de l'organisation. Sociologie et psychologie constituent deux couches, différentes mais solidaires, d'un même empilement.

5. "Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software". (voir www.gnu.org/philosophy/free-sw.html).

« The Hackers Ethic »

(Steven Levy [113] p. 38)

1) *L'accès aux ordinateurs - et à tout ce qui peut vous apprendre quelque chose sur la façon dont le monde fonctionne - doit être illimité. Respectez le « Hands-On Imperative » !* Quand on rencontre un obstacle, il faut « y mettre les mains » et résoudre le problème sans attendre que l'on vous y invite ou que l'on vous y autorise ^a.

2) *L'information doit être gratuite.* La gratuité de l'information et des logiciels est supposée plus efficace, par les synergies qu'elle permet, qu'une économie où les logiciels seraient vendus sur le marché et protégés par des copyrights.

La transition entre la gratuité et le marché sera le « software flap » provoqué par Bill Gates lorsqu'il publia en février 1976 sa « Open Letter to Hobbyists » dans *Computer Notes*, « newsletter » des utilisateurs de l'Altair (voir page 111).

3) *Ne pas faire confiance à la hiérarchie, promouvoir la décentralisation.* Les hackers étaient, bien avant d'autres, des partisans de l'organisation transverse qu'ils jugeaient seule efficace. Ils étaient par ailleurs insensibles aux prestiges de la hiérarchie, comme le montre la règle suivante :

4) *Juger les hackers selon la qualité de leurs hacks et non selon des critères farfelus comme le diplôme, l'âge, la race ou le grade.* Seule compte la compétence, le niveau atteint dans la maîtrise de la machine : il s'agit de contourner les obstacles que celle-ci oppose à ceux qui veulent la plier à leurs besoins.

5) *Vous pouvez créer de l'art et de la beauté avec un ordinateur.* L'ordinateur n'est pas seulement fait pour calculer, comme le suggère « computer », ni pour mettre de l'ordre, comme suggère « ordinateur » : on doit pouvoir l'utiliser pour faire de la musique, dessiner, créer des mondes imaginaires qui donneront aux rêves un prolongement (presque) aussi vrai que le monde réel.

C'est l'origine des mondes virtuels qui ont eu tant d'importance lors de l'explosion du marché des jeux. Il est significatif qu'une bonne part du progrès des ordinateurs en performance, convivialité et ergonomie provienne de leur utilisation ludique. Il est intéressant de noter que les « hackers de la troisième génération », qui ont grandi dans les années 1970, sont venus à l'informatique par la pratique des jeux (Bennahum [10]).

6) *Les ordinateurs peuvent améliorer votre vie.* Les hackers n'ignorent pas les risques que comportent certains usages de l'ordinateur (en particulier ses utilisations militaires les effraient), mais ils affirment que des utilisations utiles, progressistes, constructives sont possibles.

^a C'est la même attitude que résume le slogan de Nike : « Just do it ».

Les héritiers des « hackers », des pionniers des années 1960 et 1970, sont ainsi aujourd'hui ceux qui travaillent sur le langage de l'entreprise, l'organisation transverse, l'articulation du système d'information avec la stratégie, la modélisation des processus, l'urbanisation des systèmes d'information etc.

Reprenons le chemin qui a conduit du premier micro-ordinateur aux réseaux de machines d'aujourd'hui.

Un précurseur : le Kenbak-1 (1971)

John V. Blankenbaker présente le Kenbak-1 au printemps de 1971 ; cette machine, dotée d'une RAM de 256 octets, était vendue en kit au prix de 750 \$. Comme le 4004 d'Intel n'était pas encore commercialisé - il ne sera mis sur le marché qu'en novembre 1971 - l'unité centrale du Kenbak-1 comporte plusieurs composants MSI et LSI : il ne s'agit donc pas d'un micro-ordinateur au sens exact du terme, mais d'un ordinateur à circuits intégrés.

Le Kenbak-1 avait pour but d'aider l'apprentissage de la programmation. Après en avoir vendu 40 exemplaires la Kenbak Corporation a cessé ses activités en 1973.

Le premier micro-ordinateur



FIG. 3.1 – *Le Micral (1973)*

En 1971 André Thi Truong, ingénieur français d'origine vietnamienne, fonde la société R2E (Réalizations Études Électroniques). En 1972 l'INRA (Institut national de la recherche agronomique) commande à R2E un système informatique transportable. R2E conçoit alors un ordinateur fondé sur le processeur 8 bits Intel 8008⁶. Le Micral est créé en six mois. La machine a été conçue par André Thi Truong, le logiciel par Philippe Kahn (figure 3.1).

Gros comme l'unité centrale d'un PC actuel, le Micral ne dispose ni d'écran ni de clavier. La saisie se fait en binaire, en manipulant des interrupteurs. On peut lui connecter un lecteur de bandes perforées.

500 Micrals sont produits la première année ; ils sont vendus 8 450 F pièce (1 750 \$ de l'époque). En juin 1973 la revue américaine *Byte* invente

6. Sorti en avril 1972, le 8008 comportait 3500 transistors. Sa vitesse était de 108 kHz, sa mémoire adressable de 16 Ko.

le mot « microcomputer » pour baptiser le Micral : le micro-ordinateur est né.

Le Micral n'a pas connu le succès commercial ; il a été utilisé essentiellement pour automatiser des postes de péage des autoroutes⁷.

Autres précurseurs

Le Scelbi (« Scientific, Electronic and Biological ») de la Scelbi Consulting Company (Milford, Connecticut) fut annoncé dans l'édition de mars 1974 de la revue *QST*. Construit autour du microprocesseur 8008 d'Intel, le Scelbi avait 1 Ko de mémoire programmable et il était vendu 565 \$. 15 Ko de mémoire supplémentaire étaient fournis pour 2 760 \$.

Le Mark-8, conçu par Jonathan Titus, était lui aussi construit autour de l'Intel 8008. Il fut annoncé dans le numéro de juillet 1974 de *Radio Electronics*.

L'Altair 8800 (1974)

Intel sortit le microprocesseur 8080 en juillet 1974. Il était vingt fois plus rapide que le 4004 et son prix pouvait descendre jusqu'à 3 \$ pour des commandes en quantité. Ce microprocesseur sera au cœur de la conception de l'Altair, la machine qui a enfin fait démarrer le marché du micro-ordinateur.

MITS (« Micro Instrumentation Telemetry System »), petite entreprise d'Albuquerque au Nouveau Mexique dirigée par Ed Roberts, a lancé l'Altair 8800 en décembre 1974. La photographie de ce micro-ordinateur paraît pour la première fois sur la couverture du numéro de janvier 1975 de la revue *Popular Electronics* (figure 3.2). Dans les deux mois qui suivent, des milliers de commandes arrivent à MITS.

L'Altair était commercialisé en kit et il fallait beaucoup d'habileté pour le monter. L'unité de base avait 256 octets de RAM et coûtait 395 \$: ainsi l'Altair était beaucoup moins cher que le Micral. Comme il n'existait pas de logiciel, les utilisateurs devaient écrire leurs propres programmes puis les saisir en binaire en appuyant sur des boutons (des cals se formaient bientôt au bout de leurs doigts...). Le résultat était affiché en binaire sur des diodes.

On pouvait ajouter au système un bus et des cartes d'extension. En quelques mois les cartes qui permettaient d'accroître la mémoire ou de raccorder des terminaux furent disponibles. Le télétype ASR-33 fournit un clavier, une imprimante et un support de stockage sur bande de papier perforé. MITS commercialisa par la suite d'autres versions améliorées (8800a, 8800b).

7. En 1978, R2E est absorbée par Bull. En 1982, la filiale américaine de Bull concevra son premier compatible PC, le Bull Micral. Truong quittera Bull en 1983.

HOW TO "READ" FM TUNER SPECIFICATIONS

Popular Electronics

WORLD'S LARGEST-SELLING ELECTRONICS MAGAZINE · JANUARY 1975/75¢

PROJECT BREAKTHROUGH!

World's First Minicomputer Kit to Rival Commercial Models...

"ALTAIR 8800" SAVE OVER \$1000



ALSO IN THIS ISSUE:

- An Under-\$90 Scientific Calculator Project
- CCD's—TV Camera Tube Successor?
- Thyristor-Controlled Photoflashers



TEST REPORTS:

- Technics 200 Speaker System
- Pioneer RT-1011 Open-Reel Recorder
- Tram Diamond-40 CB AM Transceiver
- Edmund Scientific "Kirlian" Photo Kit
- Hewlett-Packard 5381 Frequency Counter

FIG. 3.2 – La revue qui a éveillé les premières vocations de programmeurs pour micro-ordinateur

L'Altair et la naissance de Microsoft

Les origines de Microsoft sont intimement liées à l'arrivée de l'Altair sur le marché. Suivons la chronologie telle qu'elle est décrite dans la biographie de Bill Gates⁸.

Paul Allen (né en 1953, alors employé par Honeywell) et Bill Gates (né le 28 octobre 1955, alors étudiant en deuxième année à Harvard) étaient des amis passionnés par l'informatique, des hackers. L'article de *Popular Electronics* les incita à programmer un interpréteur Basic pour l'Altair : ce sera le premier langage de programmation pour micro-ordinateur. Allen et Gates vendirent la licence de cet interpréteur à MITS le 1er février 1975.

Le 1er mars 1975, Allen est embauché par MITS comme « Director of Software ». Le 7 avril 1975, *Computer Notes*, la « newsletter » des utilisateurs de l'Altair, annonce que l'interpréteur Basic est opérationnel (*Up and running*). Une version 2.0 est publiée le 1er juillet 1975. Le 22 juillet 1975, Allen et Gates signent avec MITS un accord de licence. Le 29 novembre 1975, Gates utilise dans une lettre à Allen le nom « Micro-Soft » : c'est la première mention écrite connue de ce nom.

Le 3 février 1976, Bill Gates publie dans *Computer Notes* sa « Open Letter to Hobbyists » [62] (voir ci-dessous). Le 1er novembre 1976, Allen quitte MITS pour rejoindre Microsoft à plein temps.

Le 26 novembre 1976, la marque « Microsoft » est enregistrée auprès du bureau du secrétariat d'état du Nouveau Mexique pour « nommer des programmes informatiques, des systèmes d'exploitation et des services ». La déclaration mentionne que le nom de Microsoft a été utilisé depuis le 12 novembre 1975.

La lettre ouverte de Bill Gates aux « Hobbyistes »

Dans sa lettre ouverte (voir sa traduction page 112 ; le texte original est à la page www.volle.com/rennes/lettrebill.htm), Bill Gates accuse carrément de vol les « hobbyistes » qui ont piraté son interpréteur Basic. Il leur reproche d'empêcher la production de bons logiciels et termine par une phrase prophétique : « Rien ne me ferait plus plaisir que de pouvoir embaucher dix programmeurs pour inonder le marché de bons logiciels ».

Cette lettre est importante pour l'histoire de l'informatique : elle inaugure le modèle économique du « logiciel compilé marchand vendu en boîte » qui succédera au modèle du « programme source librement retouchable » des hackers. Bill Gates avait vingt ans lors de sa publication.

Même si le modèle « open source », revitalisé par l'Internet, est revenu en force vingt ans après la publication de cet article, le modèle marchand a eu son utilité : il était sans doute le seul dans les années 1970 qui pût permettre la production rapide des logiciels nécessaires au succès du micro-ordinateur. Accessoirement, il fut aussi à l'origine de la croissance de Microsoft et de la fortune de Bill Gates.

8. Source : www.microsoft.com/billgates/bio.htm.

3 février 1976

Lettre ouverte aux « Hobbyistes »

De mon point de vue, le manque de bon cours de programmation, de bons livres et de logiciels est aujourd'hui pour les hobbyistes la question la plus critique. Sans un bon logiciel, si l'utilisateur ne connaît pas la programmation, l'ordinateur du hobbyiste reste stérile. Écrira-t-on du logiciel de qualité pour ce marché?

Voici près d'un an, Paul Allen et moi avons anticipé sa croissance, embauché Monte Davidoff et développé le BASIC pour l'Altair. Le travail initial n'a pris que deux mois mais nous avons tous les trois dû passer l'essentiel de l'année dernière à documenter ce BASIC, l'améliorer et l'enrichir en fonctionnalités. Nous avons maintenant des versions BASIC pour les mémoires 4K, 8K, étendues, ROM et disque dur. Nous avons utilisé pour plus de 40 000 \$ de temps d'ordinateur.

Nous avons reçu des réactions positives de centaines de personnes qui utilisent ce BASIC. Cependant, deux faits surprenants apparaissent : 1) la plupart de ces « utilisateurs » n'ont jamais acheté le BASIC (il a été acheté par moins de 10 % des possesseurs d'un Altair), 2) le montant des redevances venant de nos ventes aux hobbyistes rémunère le travail fourni sur le BASIC de l'Altair à moins de 2 dollars de l'heure.

Pourquoi cela? La plupart des hobbyistes savent bien qu'ils volent le logiciel. Il faut bien acheter le matériel, mais le logiciel, ça se partage. Qui se soucie de rémunérer les gens qui ont travaillé pour le produire?

Est-ce honnête? Si vous volez le logiciel, vous ne pourrez pas vous retourner contre MITS en cas de problème. MITS ne fait aucun profit en vendant le logiciel. La redevance qui nous est payée, le manuel, la bande etc. en font une opération tout juste équilibrée. Ce que vous faites, c'est d'empêcher la production de bon logiciel. Qui peut se permettre de faire travailler des professionnels pour rien? Quel hobbyiste pourrait mettre trois hommes*année dans la programmation, détecter toutes les bogues, documenter le produit puis le distribuer pour rien? Le fait est que personne, en dehors de nous, n'a investi d'argent dans le logiciel pour les hobbyistes. Nous avons écrit le BASIC pour le 8080, nous sommes en train d'écrire l'APL pour le 8080 et le 6800, mais rien ne nous incite à mettre ces logiciels à la disposition des hobbyistes. Vous êtes tout simplement des voleurs.

Que penser de ceux qui revendent le BASIC pour Altair? Ne se font-ils pas de l'argent sur le marché des logiciels pour hobbyistes? Oui, mais ceux que l'on nous a signalés pourraient finalement y perdre. Ils donnent une mauvaise réputation à tous les hobbyistes: on devrait les chasser des réunions des clubs où ils apparaissent.

J'aimerais recevoir des lettres de tous ceux qui souhaitent payer leur dette envers nous, ainsi que de ceux qui ont des suggestions ou des commentaires à formuler. Écrivez-moi à 1180 Alvarado SE, #114, Albuquerque, New Mexico, 87108. Rien ne me plairait davantage que de pouvoir embaucher dix programmeurs pour inonder de bons logiciels le marché des hobbyistes.

Bill Gates
Directeur Général, Micro-Soft

3.1.1 Les apports du PARC de Xerox

Les dirigeants de Xerox, entreprise qui avait construit sa fortune sur le marché des photocopieurs avec la Xérographie, ont eu l'intuition du caractère porteur de la microélectronique et de l'informatique. Xerox avait acheté Scientific Data Systems en 1969⁹; elle décida alors de créer un centre de recherche.

Le « Palo Alto Research Center » (PARC) de Xerox est créé en 1970. Il sera situé dans la Silicon Valley, tout près de l'Université de Stanford, afin de pouvoir bénéficier des apports de la recherche universitaire et de la force de travail des étudiants. Quelques-uns des hackers les plus créatifs se rassemblent alors au PARC. Ils vont apporter à l'informatique des innovations dont le micro-ordinateur tirera parti (voir la liste des principales d'entre elles page 114).

L'histoire du PARC comporte cependant un paradoxe : si Xerox a généreusement financé ce centre de recherche, elle n'a pratiquement pas tiré parti de ses découvertes qui toutes seront finalement commercialisées par d'autres entreprises (Hiltzik [82]). Xerox a construit l'Alto, mais c'est IBM qui a lancé le PC en 1981. Xerox a inventé l'interface graphique¹⁰ avec fenêtres, souris et menus déroulants, mais c'est Apple qui a lancé le Macintosh et Microsoft qui a produit Windows. Xerox a conçu le traitement de texte Wysiwyg¹¹, mais c'est Microsoft qui a produit Word. Xerox a mis au point le protocole de réseau Ethernet, mais aujourd'hui le marché des réseaux locaux se partage entre Cisco et 3Com. Xerox a inventé l'imprimante à laser, mais il s'est fait précéder sur ce marché par IBM en 1975.

L'une des dates les plus importantes de l'histoire de l'informatique est celle de la visite de Steve Jobs et d'une équipe d'ingénieurs d'Apple¹² au PARC en décembre 1979 : la démonstration à laquelle ils assisteront les incitera à introduire l'interface graphique dans le Lisa (1983) et dans le Macintosh (1984).

Pourquoi Xerox n'a-t-il pas utilisé les travaux du PARC ? Il est tentant mais trop facile d'expliquer cela par des comportements personnels, des conflits politiques dans l'entreprise ou la stupidité de bureaucrates incapables de percevoir le potentiel d'une innovation. Ces facteurs ont joué,

9. L'année même où l'Arpanet, précurseur de l'Internet, devient opérationnel (voir page 335).

10. Les éléments de l'interface graphique sont les fenêtres (*windows*), menus, icônes, boutons, onglets ainsi que le pointeur. Les fenêtres permettent la représentation simultanée de plusieurs activités sur l'écran. Les menus permettent de choisir les prochaines actions. Les icônes, boutons, onglets etc. attribuent une forme concrète à des objets informatiques. L'outil de pointage, souris ou track-ball, sélectionne fenêtres, menus, icônes etc.

11. Le Wysiwyg (« What You See Is What You Get ») est l'un des principes les plus importants de l'interface graphique : chaque manipulation de l'image sur l'écran entraîne une modification prévisible, et visible, de l'état du système.

12. Apple Computer avait été créé en avril 1976 par Steven Wozniak et Steven Jobs. Leur Apple I (1976) n'a pas été pris au sérieux par les hobbyistes, mais l'Apple II (1977), premier ordinateur personnel carrossé en plastique et présentant une interface en couleur, a connu le succès. Le premier tableur, Visicalc (1979), a été réalisé pour l'Apple II. L'Apple III sortira en 1980.

Les inventions du PARC

1971 : Alan Kay et son équipe mettent au point la première version de Smalltalk, langage de programmation orienté objet qui influencera C++ et Java ; Gary Starkweather met au point la première imprimante à laser.

1973 : Chuck Thacker, Butler Lampson et Alan Kay mettent au point l'Alto, ordinateur qui ne sera jamais commercialisé. Robert Metcalfe crée le protocole de réseau local Ethernet.

1974 : Dan Ingals invente un système qui offre l'interface avec fenêtres superposées et menus déroulants. Charles Simonyi, Tim Mott et Larry Tesler produisent le premier traitement de texte convivial.

1979 : James Clark conçoit le microprocesseur graphique 3D qui fera par la suite la fortune de Silicon Graphics.

30 septembre 1980 : les spécifications du réseau Ethernet sont publiées. C'est la première fois qu'une invention du PARC est commercialisée.

27 avril 1981 : Xerox lance la station de travail Star (16 000 \$), descendant commercial de l'Alto et des travaux du PARC. Mais le 24 août 1981 IBM lance le PC (2 000 \$) qui rend instantanément le Star obsolète.

1983 : Après la démission forcée de Bob Taylor le 19 septembre 1983, plusieurs des ingénieurs du PARC démissionnent. Le style de travail du PARC ne sera plus le même.

Janvier 1984 : Apple lance le Macintosh, incarnation réussie de l'ordinateur personnel conçu par le PARC.

mais ce ne sont pas les plus importants. En fait il n'était pas facile de réussir dans le domaine des ordinateurs personnels, comme l'ont montré les échecs rencontrés ensuite par IBM et même par Apple ; et s'il était en 1981 possible pour Apple (40 personnes) de prendre de gros risques, c'était pratiquement impossible pour Xerox (125 000 personnes).

Xerox s'était spécialisée sur le marché des photocopieurs, grosses machines que l'on installe dans les secrétariats et que le fournisseur fait rémunérer à la copie. Le succès de la Xérographie avait mis Xerox en situation de monopole et l'avait convaincue qu'il suffisait d'offrir de bons produits, mûrement conçus, pour que les clients se jettent dessus. Cela ne la préparait pas au marché de l'informatique personnelle où les acheteurs sont les directeurs informatiques, où il n'est pas question de facturer à la consommation et où la concurrence contraint à se battre pour chaque pourcentage de part de marché.

L'intérêt de Xerox pour l'innovation était sincère mais affaire de principe plus que de stratégie ; pour que Xerox puisse devenir un acteur sur ce marché, il aurait fallu que ce fût une entreprise nouvelle, sans histoire, sans habitudes, sans organisation, libre de se transformer comme le firent Apple et Microsoft.

Toute grande entreprise, toute organisation structurée par l'histoire répu- gne à changer et a donc tendance à refuser l'innovation. Cela explique en partie les échecs de l'informatique française : celle-ci a été poussée par une administration que l'on peut considérer comme la plus grande entreprise du monde et qui, comme toute grande entreprise, est corsetée par son histoire.

Pour concevoir le PC, IBM a dû créer en son sein une organisation indépendante que le président a protégée contre le reste de l'entreprise et cela n'a pas encore suffi : la culture d'entreprise ayant rejeté cette greffe, IBM n'a pas pu profiter du PC pour dominer le marché de la micro-informatique. Ce sont Intel et Microsoft, structures minuscules en 1981, qui ont rafflé la mise.

3.1.2 IBM et le PC

La relation entre IBM et le PC a obéi à un déterminisme aussi implacable que celui d'une tragédie grecque : une entreprise dont la capitalisation boursière dépasse celle des entreprises allemandes, dont la croissance a apporté le bien-être à des vallées entières près de New-York et de Boston, fière de ses traditions de qualité et de sérieux, trébuche sur le micro-ordinateur en raison même de ces traditions et manque de s'effondrer (Carroll [33]).

* *

En 1980, le micro-ordinateur est un défi pour IBM. Le PET (Personal Electronic Transactor) de Commodore, le TRS-80 de Radio Shack et l'Apple II, sortis tous trois en 1977, séduisent des clients. Frank Cary, président d'IBM, estime que son entreprise doit être présente sur ce marché. Le micro-ordinateur étant une petite machine, sa réalisation ne devrait pas poser de problème à la plus grande entreprise informatique du monde !

Mais IBM, habituée aux gros systèmes, ne parvient pas à s'intéresser au micro-ordinateur. Le 4 juillet 1980, Cary pique un coup de sang et décide de s'occuper directement du problème. Le micro-ordinateur d'IBM sera réalisé par une équipe installée à Boca-Raton, en Floride, et séparée du reste de l'entreprise. Elle rapportera directement au président qui la financera lui-même et la protégera.

Le président veut son micro-ordinateur : il l'aura. IBM sort son premier PC (« Personal Computer ») en août 1981. C'est une réussite technique et le succès commercial est immédiat. Certains clients étaient en effet restés réticents devant le micro-ordinateur qu'ils croyaient peu sérieux. Ces réticences tombent lorsque IBM lui-même, temple du sérieux, cautionne le micro-ordinateur en produisant le PC. Il s'ensuit une avalanche de commandes.

Pour sortir le PC rapidement IBM avait sous-traité des parties essentielles à d'autres entreprises : le microprocesseur à Intel, le système d'exploitation MS-DOS à Microsoft¹³.

Le ver était ainsi dans le fruit. Le micro-ordinateur étant étranger à sa culture, IBM a manqué de flair et pris une disposition catastrophique au plan stratégique : il n'a pas introduit de clause d'exclusivité dans ses contrats avec Microsoft et Intel. Cela permettra à des fabricants de s'approvisionner auprès de ces entreprises et de produire à partir de 1986 des micro-ordinateurs « compatibles PC », des « clones », puis d'engager contre IBM une guerre des prix qui le contraindra à réduire sa marge sur les PC alors que le marché des *mainframes*, qui procurait jusqu'alors à IBM l'essentiel de son profit, souffrait de la concurrence des PC.

Les années 1980 seront pour IBM une descente aux enfers : baisse des résultats et du cours de l'action, compression des effectifs etc. Intel et Microsoft, en position de monopole sur un marché en croissance rapide, feront par contre fortune.

Certains, comme Paul Carroll dans *Big Blue* [33], ont relevé avec une ironie mordante les dysfonctionnements d'IBM ; d'autres, comme Gérard Dréan [51] expliquent les difficultés qu'a rencontrées IBM par l'intensité des changements que nous venons de décrire. Il est certain que mieux une entreprise était adaptée à l'informatique du début des années 1980, plus il lui était difficile de s'adapter à l'informatique de la fin des années 1980 (voir page 123), d'autant plus qu'elle était plus grosse, la complexité et la rigidité croissant avec la taille de l'entreprise.

13. La coopération avec Microsoft fit apparaître quelques défauts chez IBM : « Les gens de Microsoft se plaignaient de la méthode de programmation d'IBM [...] IBM mesurait le nombre de lignes produites par chaque programmeur, ce qui les encourageait à produire du code inefficace. Les gestionnaires d'IBM se plaignaient parce que, selon leur système de mesure, Microsoft ne faisait pas sa part du travail : ils disaient que si l'on comptait le nombre de lignes, Microsoft faisait en fait un travail négatif, ce qui signifiait que Microsoft aurait dû payer IBM parce qu'il condensait le code. » Cette façon de mesurer la production pousse à écrire des programmes lourds. IBM la jugeait efficace parce qu'elle induisait une forte utilisation des machines et incitait les clients à acheter de nouveaux ordinateurs plus puissants (Carroll [33] p. 101).

Il est futile d'ironiser sur les malheurs des grandes entreprises : le fait est qu'IBM, que l'on avait pu croire presque morte en 1993, s'est redressée par la suite et s'est réorientée vers le secteur des services, ce qui prouve qu'elle n'avait pas perdu son potentiel ni sa capacité d'adaptation.

3.1.3 Chronologie du PC

Voici les dates des principaux événements qui ont marqué la suite de l'histoire du micro-ordinateur :

1982 : Compaq commercialise le premier micro-ordinateur portable (15 kg !).

1983 : Apple lance le Lisa, premier ordinateur présentant une interface graphique (menus déroulants, fenêtres, corbeille etc.) : le Lisa a été inspiré par les travaux du PARC.

La norme IEEE 802.3 pour les réseaux locaux Ethernet est publiée. Elle concrétise les spécifications produites par le PARC en 1980. C'est l'amorce de la généralisation des réseaux locaux dans les entreprises.

1984 : Lotus sort le tableur Lotus 1-2-3 (voir page 138 l'histoire du tableur) qui incitera les entreprises à acheter des PC. Apple commercialise le Macintosh qui sera le grand concurrent du PC : les utilisateurs se partageront entre partisans de l'une ou l'autre des deux machines.

1985 : IBM lance le PC AT, qui a un grand succès. IBM annonce en octobre le réseau Token Ring, qui concurrencera Ethernet dans les entreprises industrielles (voir page 354).

Intel lance le même mois le processeur 80386 à 16 MHz qui améliore de façon significative la puissance du PC. Microsoft livre en novembre Windows 1.0 qui apporte l'interface graphique aux utilisateurs du PC.

1986 : Les bases de données sur PC se développent avec dBase d'Ashton et Tate. Microsoft lance le tableur Excel (d'abord connu sous le nom de Multiplan).

DEC connaît sa meilleure année, mais ce sont les derniers feux du mini-ordinateur, dont le marché est coincé entre la gamme des *mainframes* et le micro-ordinateur.

Compaq lance le marché des clones du PC en produisant le premier PC 80386. L'utilisation de la messagerie électronique se développe aux États-Unis : MCI et CompuServe offrent un lien entre leurs messageries respectives.

1987 : Les PC 80386 détrônent les PC AT. IBM lance la série PS/2 et le système d'exploitation OS/2. Apple lance le Macintosh II.

Le 12 juillet 1987, Robert Solow formule dans la *New York Review of Books* son célèbre paradoxe : « You can see the computer age everywhere but in the productivity statistics¹⁴ » (voir page 181).

1988 : Unix gagne en notoriété et érode la confiance dans les mini-ordinateurs et les *mainframes*. Apple poursuit Microsoft et HP en justice à propos de l'interface du PC. Compaq prend la tête d'un consortium de fournisseurs connu comme « gang des neuf », et crée le standard EISA pour

14. « On voit des ordinateurs partout, sauf dans les statistiques de productivité. »

contrer le Micro Channel du PS/2 d'IBM. Il annonce le mini-ordinateur AS/400.

1989: Ethernet 10BaseT démarre. C'est l'année des réseaux locaux de PC; les hubs et adaptateurs de SynOptics et 3Com ont un grand succès. Cela prépare la voie aux applications client/serveur des années 1990. En avril Intel annonce le processeur 80486. IBM sort OfficeVision, sa première suite d'applications. Apple lance un Macintosh portable (7 kg!).

1990: C'est l'année des routeurs et des WAN (« Wide Area Network ») interconnectant les réseaux locaux. Microsoft lance Windows 3.0. En septembre, IBM et Microsoft redéfinissent leur partenariat: IBM prend la responsabilité d'OS/1.x et 2.x, Microsoft celle de l'OS/2 portable, de DOS et de Windows. Motorola lance le processeur 68040, Apple lance des Macs bas de gamme: Classic, LC et IIsi.

1991: Windows est en position de monopole et OS/2 disparaît de la scène. Naissance du *World Wide Web*: Tim Berners-Lee, au CERN de Genève (« Conseil Européen pour la Recherche Nucléaire »), monte le premier serveur Web (voir page 349).

Les dépenses des entreprises américaines en informatique dépassent pour la première fois les dépenses en équipement industriel, agricole et en construction. Apple lance la première génération de Powerbooks.

1992: C'est l'année des applications sur réseau local et aussi celle du groupware avec Lotus Notes. L'*outsourcing* démarre avec un gros contrat passé par Kodak.

IBM et Microsoft mettent fin à leur accord de coopération. Pour la première fois, les comptes annuels d'IBM font apparaître une perte de 564 millions de dollars. IBM lance le premier de ses Notebooks: le ThinkPad.

1993: Début du déploiement du Pentium. Les pertes d'IBM sont les pires de son histoire: 4,97 milliards de dollars pour un chiffre d'affaires de 64,5 milliards. Lou Gerstner remplace John Akers à la tête d'IBM: c'est le premier « outsider » qui prend le poste de CEO.

Apple perd son procès contre Microsoft et HP. Intel lance le processeur 60 MHz Pentium, Apple sort le Newton, Novell annonce NetWare 4.0, Lotus Notes 3.0 démarre, Microsoft lance Windows NT.

1994: L'architecture client/serveur devient à la mode.

L'erreur de calcul révélée sur la puce Pentium suscite un drame de relations publiques pour Intel, qui y met un terme en remplaçant toutes les puces.

Microsoft annonce que Windows 95 ne sera pas livré avant août 1995, ce qui mécontente beaucoup d'utilisateurs contraints à revoir leurs plans de migration vers le 32 bits.

La frénésie des fusions et acquisitions continue: Novell achète WordPerfect (voir page 134 l'histoire du traitement de texte), Aldus et Adobe fusionnent. Apple entre sur le marché de la vente en ligne avec eWorld. Netscape, le futur chéri de Wall Street, fait ses débuts.

1995: Les Notebooks deviennent une alternative au desktop avec les portables Pentium. Il en résulte un développement du télétravail.

IBM fait une offre de 3,5 milliards de dollars pour acheter Lotus.

En août, Microsoft livre Windows 95 et Intel lance le Pentium Pro à 150-200 MHz. CompuServe, AOL et Prodigy commencent à offrir des accès au Web, et Netscape lance la troisième plus importante augmentation de capital sur le Nasdaq.

1996 : Windows 95 confirme son emprise sur le PC. Windows NT gagne du terrain contre NetWare comme plate-forme pour serveurs. Les Network Computers se concrétisent par de vrais produits.

Les Intranet d'entreprise se répandent. Java gagne en notoriété. Les entreprises commencent à programmer des sites Web. Microsoft adopte enfin la Toile.

1997 : C'est l'année de l'Intranet. Le commerce électronique démarre. La navigation sur la Toile devient facile avec des navigateurs et des outils de recherche améliorés.

La puissance de traitement s'accroît lorsque Intel annonce le Pentium 200 MHz avec la technologie MMX.

1998 : La perspective de l'an 2000 effraie tout le monde. Le manque de personnel en informatique devient aigu. L'outsourcing et les services s'épa nouissent.

Le grand thème à la mode est le commerce sur l'Internet.

Plusieurs événements importants non reliés au Web se produisent : achat de Digital par Compaq, durcissement de la bataille entre Microsoft et le ministère de la justice.

3.2 Points de repère

3.2.1 Innovations importantes

Le CD-ROM

Les CD-ROM ont été initialement conçus pour porter 74 minutes de son de haute qualité. Transformés en supports de données (650 Mo), ils épargnent aux utilisateurs les heures de travail qu'ils devaient auparavant consacrer à charger une application disquette après disquette.

Les CD-ROM ont pratiquement remplacé les disquettes, d'autant plus qu'il est devenu possible de graver des CD. Le CD-ROM est cependant concurrencé par les DVD et par des disques optiques capables de contenir 4,7 Go.

Le chiffrement

Sans les algorithmes de chiffrement (voir page 369), le commerce électronique n'aurait pas pu se développer sur l'Internet : le chiffrement est crucial pour identifier les parties impliquées dans une transaction et sécuriser celle-ci.

PGP (« Pretty Good Privacy »), de Philip Zimmermann, a suscité une accélération avec son architecture ouverte et ses techniques mathématiques robustes. Les autres leaders dans ce domaine ont été RSA Data Security,

qui a testé les limites des techniques de chiffrement, et Certicom avec des innovations comme la courbe de protection elliptique (économe en largeur de bande) qui fait partie de l'offre PalmNet de 3Com.

Ethernet et les réseaux locaux de PC

NetWare 2.11, Ethernet 10BaseT et 10Base2 ont apporté aux entreprises le partage des données et de la puissance de traitement. NetWare a transformé le micro-ordinateur en une machine analogue à un *mainframe*, Ethernet a permis de relier les ordinateurs entre eux. Sans la synergie entre ces deux techniques, l'explosion de l'Internet n'aurait pas pu se produire car le réseau d'ordinateurs serait resté coûteux. Elles sont évolutives : Ethernet devient plus rapide et s'étend à de nouveaux médias, les réseaux locaux s'interconnectent.

L'Intranet

Cela commence par les systèmes de messagerie *store and forward* des années 1960 et 70 (à distinguer du *store and retrieve*) et par les conférences sur *mainframe* et mini, particulièrement dans les universités et le monde de la recherche.

Puis la mise en réseau des PC et la normalisation des protocoles répandent la communication dans l'entreprise et entre les entreprises. cc:Mail de Lotus et MHS (« Message Handling Service ») de Novell aident à disséminer la messagerie alors que le groupware, avec Lotus Notes, fournit des outils de programmation personnalisée et un lien avec les applications externes.

Les années 1990 et 2000 sont celles du temps réel : le groupware a intégré les communications synchrones (conversation, visioconférence, partage d'applications). Ces fonctionnalités seront finalement absorbées par l'Intranet qui s'installera dans les entreprises dans la foulée de la Toile.

Le Macintosh

En lançant le Macintosh en 1984, Apple a transformé pour toujours les ordinateurs en introduisant trois avancées technologiques (déjà utilisées dans la ligne de produits Lisa) : l'interface graphique ; la souris ; les disquettes 3.5 pouces à haute capacité et très solides.

La conception du Lisa et du Mac doit beaucoup aux idées du centre de recherche de Xerox à Palo Alto, mais Apple les a retravaillées pour les industrialiser et les rendre plus commodes pour l'utilisateur.

En refusant d'adapter son système d'exploitation MacOS à d'autres processeurs que la famille 68000 ou d'en vendre la licence à d'autres entreprises, Apple a contraint les fabricants de clones à chercher un autre système d'exploitation. Microsoft s'est engouffré dans la brèche et, comme Apple perdait du temps en explorant trop de sentiers différents, Windows a fini par devenir le système d'exploitation préféré des entreprises et des utilisateurs.

Toutefois la migration vers le microprocesseur à architecture RISC d'IBM pour le PowerPC, puis la sortie de la ligne de produits iMac ont maintenu Apple à flot et lui ont permis de rebondir.

Le logiciel libre

Frustrée par les logiciels en boîtes chers et qui ne tenaient pas les promesses publicitaires, la communauté des informaticiens est revenue vers le

milieu des années 1990 vers le modèle du « logiciel libre » qui permet à l'utilisateur d'accéder aux parties intimes du système.

Microsoft tâonnait pour définir sa stratégie envers la Toile et Solaris était trop cher : les entreprises ont commencé à utiliser des serveurs Web Apache sous Linux.

La RAD

Au début, la RAD (« Rapid Application Development ») n'a fait que resserrer le processus « édition - compilation - débogage » qui représentait longtemps l'essentiel du travail des programmeurs. Dans le milieu des années 1980, des produits comme Turbo Pascal de Borland ont accéléré le développement d'applications pour PC sous DOS.

Visual Basic de Microsoft, lancé en 1991, a défini un standard pour la commodité du programmeur, si ce n'est en performance ou robustesse des applications. Avec la diffusion de composants logiciels diversifiés, Visual Basic a accéléré une transition que des langages plus élégants avaient seulement promis, ouvrant la voie aux classes Java réutilisables.

L'architecture RISC

L'architecture RISC (« Reduced Instruction Set Computer ») pour les microprocesseurs, introduite par IBM sur les PC RT en 1986, a permis aux puces d'atteindre les sommets de la performance informatique. Elle était conçue pour faire plus vite les opérations habituelles et faciliter l'utilisation du microprocesseur. Mais ceux qui proposaient le RISC ont sous-estimé la solidité de la part de marché prise par l'architecture x86 d'Intel, consolidée par la largeur de la base installée en logiciels, outils et compétences.

Alpha de DEC, SPARC de Sun et PowerPC de Motorola ont tenté de prendre l'avantage sur Intel au plan de la vitesse de traitement, puis Intel introduisit les techniques RISC dans le cœur de sa ligne x86. Les puces Pentium, AMD et autres appliquent les principes RISC à l'optimisation interne tout en restant compatibles avec les versions anciennes.

VGA

On avait déjà fait de la couleur sur PC, mais l'inclusion du VGA (« Video Graphics Array ») dans la ligne PS/2 par IBM en 1987 fut un événement important. Le PC passait de 16 à 256 couleurs avec une résolution de 320 pixels par 200 ; on pouvait aussi faire du 16 couleurs avec une résolution de 640 pixels par 480. Cela favorisa le lancement des interfaces graphiques, de l'édition et des jeux d'arcade sur PC. VGA est désormais à la base de tout adaptateur vidéo.

La Toile

Quand Tim Berners-Lee appliqua pour la première fois l'hypertexte à l'Internet et forgea le terme « World Wide Web » en 1990, il cherchait à créer une collaboration facile dans les projets (voir page 349). Il a transformé l'Internet universitaire en un média de masse.

Dans les neuf années qui suivirent le premier navigateur et le premier serveur au CERN la Toile a acquis l'ubiquité. L'information et la publication furent ses premiers points forts, renforcés à la fin de 1993 par Mosaic et son utilisation graphique.

Enfin le milieu des affaires perçoit vers 1996 le potentiel de la Toile et cherche à en tirer parti. L'accroissement de la sécurité a réduit les réticences des clients envers l'achat sur le réseau, tandis que des techniques comme XML facilitent l'utilisation de la Toile par les applications professionnelles.

Windows

Windows est un exemple de la méthode pragmatique qui a permis le succès de Microsoft. D'abord pâle imitation de l'interface graphique du Macintosh, Windows était plein de bogues et handicapé par le DOS sous-jacent. Toutefois Microsoft a montré que la compatibilité ascendante, l'attention à des besoins peu intellectuels mais tirés par la productivité, le support aux programmeurs et l'OEM (« Original Equipment Manufacturer ») sont plus importants pour le succès d'un système d'exploitation que ses qualités purement techniques.

Il en résulta une adoption massive de Windows sur les PC. Microsoft appliqua en 1993 la même formule pour faire adopter Windows NT par les serveurs et stations de travail, ce qui a mis Novell et Unix sur la défensive.

3.2.2 Réussites exemplaires

Federal Express

En 1984, FedEx a lancé « Supertracker », qui procure la traçabilité aux paquets qu'il transporte. Ce système permettait aux coursiers d'utiliser des outils communicants manuels et un réseau radio pour saisir le statut et la position des paquets.

Les clients ont pu connaître par appel téléphonique, puis par la Toile, la localisation de leurs paquets et la date de livraison attendue.

Le Groupware

En 1989, les responsables du système d'information de Price Waterhouse ont fait un choix audacieux : ils ont supprimé la messagerie électronique qu'ils venaient d'installer sur 10 000 postes et l'ont remplacée par Lotus Notes, produit nouveau que personne ne comprenait à l'époque et qu'ils déployaient sous OS/2. Price Waterhouse fut ainsi la première grande entreprise à déployer Lotus Notes au niveau mondial. Cela lui a permis de gérer ses projets globalement et de gagner des appels d'offre.

Le groupware a décollé dans les entreprises de service. Coopers & Lybrand - qui a fusionné avec Price Waterhouse en 1998 - avait installé Lotus Notes en 1993, ce qui a facilité la fusion des deux entreprises.

L'interface graphique

En 1990, au milieu d'une grave crise du transport aérien, United Airlines a jugé rentable de remplacer ses terminaux « bêtes » par des PC. Jugeant Windows trop instable, United a développé sa propre interface graphique. Le système, installé sur des PC 80236, a permis de réduire le délai de traitement d'une réservation de 10 %, la durée de la formation de 25 %, et d'économiser 9 millions de dollars dès la première année.

Le Datawarehouse

Au milieu des années 1980 Wal-Mart voulait s'étendre au delà du Middle West. Mais comment gérer plusieurs magasins en tenant compte des particularités de chaque marché local? Les dirigeants de Wal-Mart eurent l'idée de collecter les données sur les ventes et de les utiliser pour répondre rapidement aux changements de tendance de la demande.

Il en résulta un des plus grands succès du datawarehouse. La base de données grimpa rapidement à 700 Go. Elle permit au détaillant de partager l'information sur la demande avec ses fournisseurs et de leur faire gérer ses stocks. Wal-Mart surpassa ainsi Sears et Roebuck. Son datawarehouse, exploité sur du matériel NCR Teradata, a crû jusqu'à atteindre la taille de 24 To.

Le commerce électronique

En 1995, Barnes & Noble et Crown Books développaient des réseaux de librairies concurrents. Jeffrey Bezos, informaticien chez Bankers Trust à Chicago, se lança sur l'Internet pour créer Amazon.com : ce sera la première grande réussite du commerce électronique.

Amazon a montré que l'on peut faire du commerce électronique malgré les inquiétudes concernant la sécurité sur le réseau. Elle offre plusieurs millions de titres de livres en ligne, des CD, des livres audio, des DVD et des jeux pour ordinateur.

Par ailleurs Dell a prouvé qu'il était rentable pour la distribution des micro-ordinateurs d'intégrer le commerce électronique avec une chaîne d'approvisionnement. Lancé au début de 1997, le site de Dell donne aux clients des dates de livraison exactes en contrôlant les stocks de pièces détachées lors de la commande. Dell partage les informations sur la demande avec ses fournisseurs en temps réel, ce qui lui permet de réduire les coûts d'immobilisation et d'améliorer le service au client.

3.3 Le marché de l'informatique

La vivacité de la concurrence entre les fournisseurs contribue à expliquer la baisse du prix des ordinateurs (voir page 55). Les progrès des systèmes d'exploitation, applications, processeurs et mémoires offrent autant d'occasions pour relancer la concurrence, différencier les produits (tout en restant compatible avec le standard PC) et se tailler de nouvelles niches de marché.

En 1949, une étude de marché d'IBM restée célèbre avait estimé que neuf SSEC pourraient satisfaire les besoins des États-Unis pendant plusieurs décennies¹⁵. On dénombrait en 2000 plus de 500 millions d'ordinateurs dans le monde et depuis 1995 il se vend chaque année plus de PC que de téléviseurs. L'amélioration des techniques alimente une baisse du prix qui suscite l'élargissement du marché; cet élargissement provoque une nouvelle baisse de prix etc. L'offre se diversifie, du « desktop » au « laptop » portable,

15. Le SSEC, construit en 1948 et rival de l'ENIAC, fut l'un des premiers ordinateurs à respecter l'architecture de von Neumann (*Perspectives of the Founding Fathers* [1]).

puis au « palmtop » qui tient dans le creux de la main et qui, muni d'une antenne, apporte toutes les fonctionnalités du PC en réseau.

* *

Le micro-ordinateur a transformé dans les années 1980 le marché de l'informatique (Grove [76]). Au début de cette décennie, des entreprises comme IBM, DEC, Wang ou Univac étaient organisées verticalement, chacune étant présente dans toutes les couches (puces, ordinateurs, systèmes d'exploitation, applications, vente et distribution). L'industrie informatique était « propriétaire », un système entier étant produit par un seul fournisseur (figure 3.3). L'avantage d'une telle organisation du marché, c'est que l'offre d'un fournisseur constitue un ensemble cohérent. L'inconvénient, c'est qu'une fois le fournisseur choisi le client est contraint de lui rester fidèle : s'il voulait prendre un autre fournisseur il devrait changer tout son système informatique d'un coup, ce qui est pratiquement impossible. La compétition pour la première vente à un client est donc féroce.

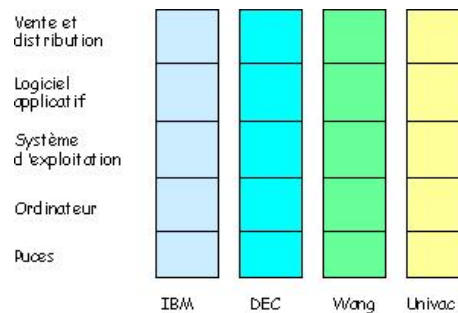


FIG. 3.3 – Partage du marché en 1980

En 1995, la structure du marché n'est plus la même. La baisse des prix a fait entrer l'informatique dans l'ère de la production de masse et le client peut faire intégrer des éléments (puce, système d'exploitation, applications) fournis par des entreprises différentes. De nouvelles compagnies se sont créées (Intel, Microsoft, Compaq etc.), spécialisées chacune dans une couche où elles sont en concurrence avec d'autres entreprises spécialisées (figure 3.4).

Ainsi pendant les années 1980, et sans que l'on puisse assigner de date précise au changement, le marché s'est restructuré. IBM, entreprise phare du début des années 1980, a subi une crise profonde ; les premières places ont été prises par les Microsoft, Intel, Compaq, Dell, Novell etc.

Les stratégies diffèrent selon le secteur d'activité. Les fournisseurs de matériel font l'essentiel de leur profit sur leurs produits les plus récents : ils souhaitent donc que l'innovation rende le matériel rapidement obsolète afin que les clients soient tentés de le renouveler fréquemment, et ils voudraient voir les fournisseurs de logiciels utiliser à fond les dernières possibilités qu'offre le matériel.

Mais les fournisseurs de logiciel s'y refusent, car seuls quelques millions d'ordinateurs disposent des dernières possibilités. Pour pouvoir toucher un large marché ils ont plutôt intérêt à produire des programmes qui pourront tourner convenablement sur les machines anciennes : ils sont donc nécessairement plus « conservateurs » que les fournisseurs de matériel (Colwell [39] p. 33).

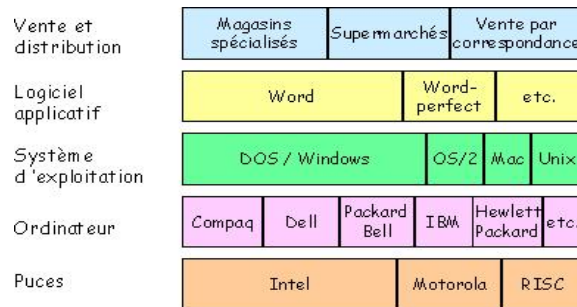


FIG. 3.4 – Partage du marché en 1990

3.3.1 Dynamique des équilibres

Pour comprendre la succession des équilibres sur le marché des logiciels, il faut remonter jusqu'aux années 1960 puis dater le passage d'un équilibre à l'autre. Le repérage de telles dates charnières est délicat : nous identifierons les événements caractéristiques sans chercher la précision des dates.

Dans les années 1960, la pratique du logiciel libre, à la fois gratuit et ouvert (le code source est mis gratuitement à disposition) est répandue sans être théorisée. Elle est mise en œuvre par des « hackers » (voir page 105).

On peut dater de 1969 la naissance du marché du logiciel. Jusqu'à cette date, IBM vendait conjointement le matériel *avec* le logiciel qui permettait de le faire fonctionner (systèmes d'exploitation, langages de programmation, applications). En 1969, pour éviter que les concurrents ne bénéficient gratuitement des logiciels qu'il avait produits - et aussi pour se prémunir contre des procès pour cause de monopole - IBM mit en place la politique d'*unbundling* (dépaquetage) qui introduisait une tarification séparée du matériel et du logiciel.

Par ailleurs à partir de la publication de la lettre ouverte de Bill Gates en 1976 (voir page 111) le commerce du logiciel pour PC compilé et vendu comme une boîte noire s'est développé.

On peut dater de 1991 - date de la première version de Linux, mais aussi date de naissance du World Wide Web - la résurgence du modèle du logiciel libre (voir www.gnu.org/philosophy/free-sw.html). Il se développera en s'appuyant sur l'Internet et en tirant argument des inconvénients que présente le logiciel compilé.

Entre 1976 et 1991 le logiciel libre n'avait pas disparu : il a constitué un thème de militantisme pour des personnes comme Richard Stallmann

(Stallman [193]) ou Eric Raymond (Raymond [171]). Cependant s'il avait un rayonnement intellectuel sa viabilité économique semblait douteuse.

Il n'en est plus de même aujourd'hui : le modèle du logiciel libre a gagné en crédibilité durant les années 1990, il a conquis des parts de marché¹⁶ et son avenir est garanti. Il reste à situer la ligne selon laquelle le marché se partagera entre les deux modèles.

3.3.2 Naissance du marché du logiciel compilé

L'*Unbundling* d'IBM

Le *consent decree* de 1956 entre IBM et le département de la justice oblige IBM à offrir à la vente les matériels qu'il louait auparavant, à facturer séparément les pièces détachées et la maintenance et à publier l'information sur la maintenance de ses machines de telle sorte qu'elle puisse être exécutée par d'autres entreprises.

Des entreprises se créent alors : elles achètent des machines IBM pour les vendre en *leasing* à un prix inférieur au loyer que pratique IBM. Pour les contrer IBM accélérera le rythme d'introduction des nouvelles machines.

Par ailleurs la publication des informations sur la maintenance avait rendu visibles les interfaces des machines IBM. RCA, Amdahl et Fujitsu ont alors pu offrir des ordinateurs compatibles avec les machines IBM et de nombreuses entreprises ont offert des périphériques compatibles (écrans, lecteurs de disques, lecteurs de bandes et imprimantes).

IBM offrait par ailleurs à ses clients ses programmes et services d'ingénierie sans les facturer séparément : leur coût était inclus dans le prix du matériel, l'ensemble constituant un « paquet » (*bundle*). Cette formule simplifiait la vie des clients mais les acheteurs de matériels compatibles disposaient gratuitement de ces services dont ils n'avaient pas supporté le coût.

Dès 1964, le Spectra 70 de RCA a été perçu comme une menace pour l'IBM 360 ; mais « les ingénieurs de RCA n'ont pas vu l'importance de la compatibilité et n'ont pas su résister à la tentation d'introduire des "améliorations" par rapport au S/360. Il en résulta que les machines de RCA ne pouvaient pas utiliser sans modification les programmes en S/360. Cette conversion étant difficile, peu de clients achetèrent les systèmes de RCA » (Humphrey [86] p. 59).

Il semblait impossible de protéger les logiciels par des brevets ; les experts d'IBM conçurent donc en 1966 un système associant copyright et licence d'utilisation.

En janvier 1969, le département de la justice engagea des poursuites contre IBM en application de la loi antitrust. L'*unbundling* fut annoncé

16. Notamment le marché des serveurs Web : en 2000, 60 % des serveurs utilisaient Apache contre 20 % à Microsoft IIS et 7 % à Netscape Enterprise. Le nombre des utilisateurs de Linux dans le monde se situerait à la même date dans la fourchette de 7 à 16 millions.

par IBM en juin 1969 : moyennant une réduction de 3 % du prix du matériel, les logiciels seraient vendus désormais à part. Par la suite, lorsque des entreprises attaqueront IBM en justice en s'appuyant sur les dispositions antitrust, IBM gagnera tous les procès en tirant argument de l'*unbundling*.

L'*unbundling* a transformé le commerce du logiciel et de l'ingénierie. IBM Global Services est devenue la partie la plus importante et la plus profitable d'IBM. L'interface de programmation (API) s'est stabilisée pour répondre aux besoins de compatibilité des utilisateurs, et cette stabilité a favorisé la diversification de l'offre.

* *

Le marché du logiciel compilé pour PC, utilisable mais illisible pour l'utilisateur, démarre avec la « Open Letter to Hobbyists » publiée par Bill Gates le 3 février 1976 (voir page 111).

Bill Gates avait alors vingt ans. Ce jeune homme avait un fort potentiel, comme on l'a vu par la suite : d'une part sa compétence en informatique faisait de lui un bon hacker ; d'autre part - et contrairement à la majorité des autres hackers - son milieu social l'avait initié aux affaires : son père était l'un des plus grands avocats d'affaires de Seattle ; sa mère siégeait au conseil d'administration de plusieurs grandes entreprises et universités.

Il était ainsi mieux préparé que d'autres à percevoir le potentiel économique du logiciel compilé, et peut-être aussi à comprendre que seule une telle organisation du marché pourrait fournir la diversité des logiciels dont les micro-ordinateurs allaient avoir besoin.

Les hackers sont restés désarmés devant son attaque. Gates les avait pris en tenaille entre deux cultures américaines : celle du pionnier qui va de l'avant dans des territoires vierges et s'y sert de l'*open source* pour se débrouiller ; celle de la libre entreprise, qui ne peut pas se concevoir sans une protection du droit de propriété. C'était un nouvel épisode de la lutte entre les agriculteurs et les éleveurs, entre les fermiers et les cow-boys !

En définissant le modèle économique qui s'imposera sur le marché des logiciels pour PC, Gates a créé une industrie dont il est devenu le plus grand dirigeant. Le modèle qu'il a inventé était sans doute alors le seul possible et sa fécondité suscite l'admiration, quelle que soit l'opinion que l'on peut avoir sur la qualité des produits de Microsoft.

Cependant ce modèle a une limite qui se révèle avec la complexification des logiciels. Elle est due notamment aux exigences de la « compatibilité ascendante » : la version V_n d'un logiciel doit en principe être capable de lire et de traiter les fichiers composés avec les versions V_{n-k} antérieures. En passant d'une version à l'autre, le logiciel s'alourdit de conventions anciennes dont il doit garder la trace.

La complexification est due aussi à la cible marketing elle-même : pour que le logiciel puisse couvrir un large marché il doit offrir une grande diversité de fonctionnalités et pouvoir être exécuté sur des plates-formes diverses. Chaque client n'utilisera qu'une petite partie du logiciel qu'il a acheté, et dont le volume encombre la mémoire vive de son ordinateur.

Or plus un logiciel est compliqué, plus il est difficile de le déboguer. Le logiciel est une construction fragile et quand on corrige une erreur on risque d'en introduire plusieurs autres ; l'un des résultats les plus intéressants de la théorie de l'informatique, c'est qu'il est impossible d'automatiser complètement le débogage (voir page 84).

On ne peut physiquement plus désormais, même avec 30 000 programmeurs groupés sur un campus (plus quelques sous-traitants qui coopèrent à travers l'Internet) faire converger convenablement le processus de « débogage » d'un grand logiciel comme Windows. Les versions successives sont donc commercialisées avec des bogues et les utilisateurs subissent des incidents désagréables¹⁷.

Organisation de la production d'un grand logiciel

Pour réaliser une construction intellectuelle très compliquée comme un système d'exploitation ou un grand logiciel applicatif, il faut réunir les quatre éléments suivants :

- un *centre* capable d'attirer les contributeurs et de sélectionner les contributions à retenir ;
- un *réseau* de communication ;
- des *contributeurs* qui écrivent le code ;
- un moyen de *rémunérer* les contributeurs.

Le modèle « Microsoft » a permis de rassembler la masse critique de contributeurs nécessaire à la production des logiciels pour PC. Le centre, c'est Bill Gates lui-même assisté par une équipe de proches collaborateurs ; le réseau, c'est le réseau local du campus de Redmond et l'Internet entre Microsoft et ses sous-traitants ; les contributeurs sont les programmeurs de Microsoft et les entreprises sous contrat ; la rémunération est financière.

Dans le modèle « Linux », le centre est Linus Torvalds assisté par une équipe de proches collaborateurs ; le réseau est l'Internet ; les contributeurs sont potentiellement tous les programmeurs du monde entier ; la rémunération est symbolique (ce qui ne veut pas dire qu'elle soit irréaliste), ce qui implique que le centre soit un « dictateur bienveillant » (Tirole et Lerner [112]).

3.3.3 Le modèle du logiciel libre

La culture finlandaise de Linus Torvalds¹⁸ le libère des dogmes de la libre entreprise à l'américaine. Il tire parti de l'Internet pour utiliser le mode de

17. Microsoft collecte *via* l'Internet les comptes rendus d'incident et diffuse des versions corrigées que les utilisateurs peuvent télécharger : la chasse aux bogues se poursuit ainsi après la diffusion du produit.

18. Les grands créateurs de logiciels ont chacun ses racines culturelles : ainsi les écrits de Bjarne Stroustrup (C++), danois, rappellent l'existentialisme protestant de Kierkegaard ; Linus Torvalds (Linux), finlandais, a été marqué par l'épopée du Kalevala ; Niklaus Wirth

développement « open source » : le code source de Linux est disponible sur l'Internet, ainsi d'ailleurs que ceux de Java, Perl etc.

Quiconque repère une bogue peut, s'il en a la compétence, proposer une correction qui sera soumise au « dictateur bienveillant ». L'Internet élargit à des millions de programmeurs le cercle des contributeurs potentiels ; il permet d'accélérer la convergence du déboguage et de poursuivre sans fin le processus de perfectionnement. Il permet aussi à l'utilisateur de choisir « à la carte » les fonctionnalités dont il a besoin, ce qui réduit la taille des logiciels.

Se crée alors, en contraste avec l'économie marchande du logiciel compilé, une économie indirecte de la reconnaissance professionnelle autour du logiciel libre¹⁹ : « indirecte » parce que si un programmeur contribue gratuitement à Linux il sera respecté dans son entreprise et prendra de la valeur sur le marché. L'économie de la reconnaissance est certes une économie symbolique, mais non une économie de la gratuité car le symbole est rémunérateur psychologiquement et, à terme, financièrement²⁰.

D'ailleurs « logiciel libre » ne signifie pas « logiciel gratuit », mais « logiciel dont le code source est lisible ». Une économie marchande s'est bâtie sur le logiciel libre, qui se vend toutefois à des prix très inférieurs à ceux du logiciel compilé.

Le « dictateur bienveillant »

Dans le monde du logiciel libre, la production d'un gros programme ne peut aboutir que si elle est animée par un « dictateur bienveillant ». Le talent nécessaire pour jouer ce rôle est donc crucial.

L'expression « dictateur bienveillant » est bien sûr paradoxale. Il faut la comprendre ainsi : beaucoup de personnes (à vrai dire, tout le monde) peuvent suggérer une modification du code, mais seule une petite équipe est autorisée à introduire effectivement cette modification : c'est là le côté « dictateur ».

Cependant cette petite équipe manifeste de la considération à celui qui fait une suggestion, même et surtout si la suggestion n'est pas retenue : c'est cela le côté « bienveillant ».

Si le « dictateur » était désagréable ou méprisant, il ne recevrait plus de suggestion et le processus d'amélioration serait stoppé.

(Pascal), suisse, par le calvinisme ; Larry Wall (Perl) et Alan Kay (Smalltalk), tous deux américains, ont cherché leurs références dans le monde de l'enfance. Voir « Interview with Linus Torvalds: What Motivates Free Software Developers? » www.firstmonday.dk/issues/issue3_3/torvalds/index.html, 1999.

19. « The "utility function" Linux hackers are maximizing is not classically economic, but is the intangible of their own ego satisfaction and reputation among other hackers » (Raymond [171]).

20. Les mécanismes de reconnaissance sont ici semblables à ceux de la recherche universitaire.

Le retour de la logique du logiciel libre (et pas cher) mettra en péril la logique du programme compilé (cher) dès que les logiciels libres auront corrigé leurs défauts de jeunesse (voir ci-dessous). L'issue de la lutte est prévisible : la qualité est du côté du logiciel libre, qui seul permet de faire converger le débogage ou du moins accélère notablement sa convergence.

On peut compter sur le talent stratégique de Bill Gates : il saura adopter sagement le système de l'adversaire, même si pour le moment Microsoft résiste bec et ongles en tentant de convaincre les acheteurs et les administrations que le logiciel libre est dangereux et en s'appuyant sur la position de force que lui donne sa part de marché. Linus Torvalds le dit avec philosophie : « J'aurai gagné quand Microsoft se mettra au logiciel libre ».

* *

Le logiciel libre a lui aussi ses limites : il arrive qu'il se diversifie en variantes et qu'il soit déficient du point de vue de la documentation, des interfaces homme-machine ou de la compatibilité ascendante. Certains disent que le logiciel libre est destiné à des experts alors que Microsoft a eu l'intelligence de faire des produits destinés à l'utilisateur de base²¹ : en effet les administrateurs des serveurs Apache ou Sendmail doivent être plus qualifiés que l'utilisateur moyen d'un PC.

Les contributeurs qui participent activement à l'écriture d'un logiciel libre sont peu nombreux et la plupart des contributions sont très simples (ce qui ne veut pas dire qu'elles ne soient pas précieuses : l'alarme qui indique une bogue est toujours bienvenue). Ainsi pour Apache plus de 80 % des modifications proviendraient de quinze programmeurs seulement (Mockus, Fielding et Herbsleb [137]).

3.3.4 Pirates et protection

Allons-nous vers un « Pearl Harbor électronique » (Paulson [155]) ? Les pirates (terme qu'il faut préférer à « hackers » pour ne pas les confondre avec les pionniers des années 1960 et 1970) progressent, alors que la recherche sur la sécurité piétine.

Les pirates progressent *parce qu'ils utilisent les méthodes du logiciel libre* : ils échangent les codes sources de leurs programmes et en partagent les améliorations. Leur compétence s'accroît, la virulence de leurs produits augmente ainsi que le nombre et l'ingéniosité des attaques. Leur activité devient paraît-il rentable : les pirates racketteraient des entreprises vulnérables à qui ils vendraient une « protection », utilisant ainsi les recettes éprouvées de la mafia.

Pour faire face aux pirates on compte seulement 200 chercheurs qualifiés en sécurité dans les universités et entreprises américaines. Dès qu'un

21. « In every release cycle Microsoft always listens to its most ignorant customers. Linux and OS/2 developers, on the other hand, tend to listen to their smartest customers... The good that Microsoft does in bringing computers to non-users is outdone by the curse that they bring on experienced users. » (Tom Nadeau, « Learning from Linux » www.os2hq.com/archives/linmemo1.htm, 1999).

universitaire devient compétent il est embauché par une entreprise qui veut utiliser son savoir-faire pour se protéger : il en résulte une pénurie de professeurs qui explique le faible nombre de diplômés experts en sécurité. De façon paradoxale, la vigueur de la demande assèche ici la formation des compétences.

Ainsi la lutte est inégale : d'un côté les pirates tirent parti des méthodes de production du logiciel libre, construisent leur rentabilité, se qualifient et se multiplient ; de l'autre la recherche est entravée par le copyright, vidée de ses compétences par l'appel du marché et elle ne suit pas une politique d'ensemble. Une simple extrapolation de cette tendance désigne le futur vainqueur.

On observe toutefois chez les pirates une certaine retenue : ils pourraient bloquer l'Internet mais n'en font rien. La plupart d'entre eux sont des joueurs et ils ne pourraient plus jouer s'ils bloquaient leur terrain de jeu. Mais il suffirait qu'il se trouvât parmi eux un pervers, quelqu'un qui prenne plus de plaisir à nuire qu'à jouer (ou qui ait, pour des raisons quelconques, un fort intérêt à nuire) pour que le danger devienne réalité.

* *

Certaines des innovations permises par l'Internet (moteurs de recherche gratuits, échanges de fichiers audio et vidéo, création coopérative de bases de données) ont compromis l'équilibre économique des détenteurs de droits et des diffuseurs de contenus. Il en résulte un développement des techniques visant à instaurer un contrôle : *policy based routing* de Cisco et « réseaux intelligents » des opérateurs télécoms.

Les logiciels peuvent être protégés par des brevets depuis 1980 : en 1999, les brevets sur les logiciels ont représenté 14 % du nombre des brevets aux États-Unis. Le DMCA (Digital Millennium Copyright Act) de 1998 renforce la protection des données informatiques : les tentatives de décryptage sont interdites. La durée du copyright augmente, passant de 14 ans à 70 ans après la mort de l'auteur. D'après le projet d'UCITA (*Uniform Computer Information Transaction Act*), la vente des copies de logiciels pourrait être remplacée par un transfert temporaire du droit d'utilisation (Syme et Camp [30]).

Michel Gensollen [64] propose de redéfinir les droits de propriété sur les biens informationnels de telle sorte que leur coût de production puisse être équilibré par des recettes tout en protégeant la facilité de leur utilisation : il faudrait ainsi distinguer le droit d'accès, l'usufruit, le droit d'administration, le droit d'exclusion (déterminer qui a accès au bien), le droit d'aliénation. La rémunération devrait reposer sur des redevances plutôt que sur des droits de propriété. De nouveaux acteurs devraient émerger pour assurer l'animation des clubs de producteurs et le partage des droits.

Sur ce marché, l'innovation juridique suit l'innovation technique qui elle-même s'oriente vers les canaux que le droit protège le mieux. Nous sommes là au tout début d'une dialectique qui ressuscite, sur le réseau, les stratégies financières et commerciales qu'ont utilisées jadis Venise, les villes hanséatiques, les navigateurs et commerçants du Portugal et des Pays-Bas etc.

3.4 Les logiciels de bureautique

Nous utilisons chaque jour les logiciels de bureautique. D'où viennent-ils? Comment ont-ils été conçus? Quels sont les problèmes qu'ils posent?

La bureautique a longtemps été considérée avec méfiance par les informaticiens : qu'il soit possible d'utiliser l'ordinateur sans avoir à le programmer, cela ouvrait une brèche dans leur monopole et, il faut aussi le dire, cela créait un certain désordre.

Mais si l'on aborde le système d'information en adoptant le point de vue de ses utilisateurs on ne peut pas négliger la bureautique. Comprendre d'où elle vient, comment elle a été conçue, permet de mieux évaluer les conditions de sa bonne utilisation et de voir les causes de certaines erreurs.

Les épisodes d'apprentissage sont peu glorieux ; on se hâte de les oublier parce qu'on s'y est trouvé maladroit et un peu ridicule. Ils occupent pourtant une place importante dans les soucis des utilisateurs. J'ai reçu mon premier PC (un IBM PC AT) en octobre 1987 alors que je travaillais au CNET (Centre National d'Études des Télécommunications) à Issy-les-Moulineaux. Auparavant j'utilisais un terminal Scorpion relié au *mainframe* du centre de calcul pour écrire (en Fortran) et faire tourner des programmes de calcul économique. Ces programmes, je l'avoue, étaient riches en boucles et instructions GOTO : bâtis chacun autour d'une idée simple, ils étaient peu à peu devenus autant de plats de spaghettis, et j'avais peine à les maîtriser lorsqu'il fallait réaliser des variantes.

Les micro-ordinateurs étaient déjà répandus, mais n'étant pas de ceux qui se jettent sur la nouveauté je n'en avais pas éprouvé le besoin jusqu'au jour où Alain Le Diberder, du BIPE, m'a montré comment il utilisait Multiplan sur son PC : j'ai été intrigué, puis séduit par la puissance et la simplicité de l'outil.

Avoir sur son bureau non plus un terminal, mais un ordinateur complet avec mémoire et processeur alors qu'auparavant je n'avais jamais pu toucher la machine, c'était sensationnel mais très intimidant. La disquette, qui permettait d'introduire de nouveaux programmes dans mon PC ou de transférer des fichiers d'un PC à l'autre, m'impressionnait beaucoup : j'hésitais à l'utiliser car pénétrer l'ordinateur me semblait proche du sacrilège, fait que je livre à la sagacité des psychanalystes amateurs.

Je trouvais les commandes de MS-DOS compliquées et peu naturelles. Comment faire marcher, d'ailleurs, une machine que je ne savais pas programmer? Gérard Dubois m'éclaira quand il me dit « sur le PC on utilise trois logiciels : le tableur, le traitement de texte et le logiciel graphique » (à l'époque il n'y avait pas au CNET de réseau local, donc ni messagerie ni documentation électronique).

Dubois avait raison, puisque c'est bien ainsi que la plupart des utilisateurs se servaient de leur PC. Mais cette phrase éteignit en moi l'ambition de programmer mon PC et me détourna pour longtemps de l'invite de commandes²², cette fenêtre où figure le « prompt » C:\>.

22. Je n'y reviendrai que lorsque Laurent Bloch [13] m'aura incité à utiliser Scheme :

J'ai muni sur ses conseils mon PC du traitement de texte Textor, du tableur Multiplan et d'un logiciel graphique dont j'ai oublié le nom car je ne l'ai pratiquement jamais utilisé. Pour m'approprier au traitement de texte et au tableur j'ai dû passer des heures à lire les notices et à m'imprégner de conventions étranges (feuilles de style, macros, liens etc.). Regarder par dessus l'épaule des collègues m'a permis de découvrir quelques astuces (pour effacer un caractère, faut-il utiliser le retour arrière ou la touche « Suppr » ? Comment nommer les feuilles d'un tableur ? etc.).

Programmer en Multiplan m'a conduit peu à peu à abandonner la programmation en Fortran sans pour autant devenir un virtuose du tableur (j'avais conservé le terminal pour pouvoir utiliser mes anciens programmes et j'ai continué à programmer les variantes en Fortran). Le traitement de texte me permit de soulager Claude Bernard qui jusqu'alors avait tapé mes manuscrits. En m'entraînant je suis peu à peu devenu un dactylographe convenable : c'est sans doute ce qu'il y a eu de plus réussi dans mon apprentissage²³.

En janvier 1989 j'ai quitté le CNET pour créer avec d'autres ingénieurs Arcome, entreprise de conseil en télécoms. Antoine Laurès était un spécialiste des réseaux locaux. Fervent du Macintosh, il nous a convertis et nous avons pu découvrir ainsi une ergonomie proche de celle que l'on trouve aujourd'hui sur Windows : souris, menus déroulants, icônes, poubelle etc.

Arcome a relié ses Macintosh par un réseau local Appletalk. Nous avons découvert la messagerie, l'agenda partagé, le transfert de fichiers, le partage des imprimantes. C'était une expérience étonnante ! Je ne sais pourquoi, ces nouveautés-là m'ont paru plus naturelles que la disquette ; travailler à plusieurs sur un plateau paysagé m'y a peut-être aidé car l'échange d'astuces était intense. Cette modeste réalisation nous ayant donné de l'avance par rapport à nos clients, nous avons pu les conseiller en connaissance de cause.

En février 1990 j'ai quitté Arcome²⁴ pour créer Eutelis, autre société de conseil²⁵. J'y ai installé un réseau Ethernet pour relier des Macintosh. Un an après, l'entreprise s'est mise au PC pour utiliser la même machine que ses clients.

Philippe Penny, venu du CNET comme directeur à Eutelis, a apporté son expertise en bureautique communicante (ou « groupware ») et nous a fait découvrir Lotus Notes (Henry [81] ; Penny et Volle [216]). En utilisant la réplique entre les serveurs locaux, Lotus Notes permettait de partager une même documentation entre plusieurs sites, ce qui anticipait sur l'Internet : nous avons ainsi pu expérimenter l'Intranet avant que ce terme n'existe et l'installer chez nos clients.

ce dialecte de LISP est le langage qui convient si l'on veut programmer pour le plaisir (Friedman et Felleisen [56]).

23. La méthode que j'ai utilisée est décrite dans www.volle.com/travaux/dactylo.htm.

24. J'y avais découvert la recette infallible pour se brouiller avec son meilleur ami : s'associer avec lui pour créer une entreprise.

25. Eutelis était filiale d'un grand groupe. J'ai appris à Eutelis que le plus gros des trois mensonges de la vie en entreprise est « je viens de la holding pour vous aider » (les deux autres, plus véniels, sont « on vous rappellera » et « votre chèque est au courrier départ »).

En 1995 nous nous sommes intéressés à l'Internet auquel cette expérience nous avait préparés. Eutelis a utilisé la messagerie sur l'Internet et créé un site Web. En modélisant l'économie de l'Internet (Talière et Volle [210]) nous avons découvert la solidité de ce réseau et les perspectives du commerce électronique. En août 1998, m'appuyant sur cette expérience, j'ai créé le site `www.volle.com` et utilisé FrontPage pour publier mes travaux sur la Toile.

Parallèlement aux expériences que nous faisons dans notre entreprise, nous avons mis en place des systèmes d'information chez nos clients. Pour certains d'entre eux le groupware fournissait des solutions élégantes, efficaces et peu coûteuses. Il était cependant difficile d'en convaincre les informaticiens, qui se méfiaient de la bureautique et préféraient des projets plus lourds. Par ailleurs les changements qu'introduisaient dans l'organisation du travail les workflows, la rédaction coopérative, la dissémination sélective etc. (Winograd et Flores [59]) n'étaient pas toujours les bienvenus malgré leur utilité. Enfin les utilisateurs rencontraient naturellement les mêmes difficultés d'apprentissage, les mêmes inhibitions que celles que j'avais connues moi-même à mes débuts.

Pour surmonter ces inhibitions, il est utile d'enjamber la barrière qui sépare l'utilisateur des concepteurs et de voir comment ces derniers ont fait pour mettre au point des produits aussi étranges. Nous allons parcourir l'histoire de deux des plus importants d'entre eux : le traitement de texte et le tableur.

3.4.1 Histoire du traitement de texte

L'utilisateur d'un traitement de texte sur ordinateur dispose aujourd'hui d'une grande diversité de fonctions dont presque toujours il ne connaît qu'une partie. Cette accumulation s'est faite progressivement. L'histoire de WordPerfect permet de l'illustrer : à partir de l'innovation initiale, ce produit a en effet parcouru toutes les étapes de l'évolution (Peterson [158]).

* *

En 1977 le traitement de texte est un sujet neuf. Les machines de traitement de texte (comme la Wang) sont des machines à écrire dotées d'un processeur et d'une mémoire et la moins chère d'entre elles coûte 15 000 \$. Il existe aussi sur ordinateur des traitements de texte informatiques « au kilomètre » (*run-off*) : l'affichage à l'écran est hérissé de codes, il faut reformater le document avant de l'imprimer et on ne découvre son apparence qu'après l'impression.

Alan Ashton, professeur d'informatique, se lance alors pour le plaisir dans la conception d'un traitement de texte. Il avait auparavant écrit un programme de musique sur ordinateur, ce qui l'avait conduit à aborder les problèmes que pose l'affichage en temps réel. Il produisit un programme qui permettait d'afficher à l'écran l'apparence de ce qui serait imprimé et de faire défiler le texte à l'écran sans interruption et non plus page à page. Il supprima la distinction entre les modes *Edit*, *Insert* et *Create* : l'utilisateur pouvait taper partout dans le document et y insérer du texte sans avoir à changer de mode.

Ce programme, amélioré et industrialisé, deviendra WordPerfect qui sera commercialisé en 1980 par la société SSI, créée en 1978. Son prix était de 5 500 \$: il était donc économique pour une entreprise d'acheter un logiciel fonctionnant sur ordinateur plutôt qu'une machine de traitement de texte ; cependant il fallait être sûr que l'exécution de WordPerfect ne ralentirait pas l'ordinateur, utilisé alors surtout par la comptabilité.

La première version de WordPerfect était rapide et facile à utiliser mais ne fonctionnait que sur les ordinateurs de Data General, sous le système d'exploitation AOS et avec une imprimante Diabolo 1650. Durant les années suivantes, une grande part du travail de programmation sera consacrée à l'adaptation de WordPerfect aux divers systèmes d'exploitation²⁶ et imprimantes du marché.

Le 12 août 1981 IBM sort le PC. Wordstar, le premier traitement de texte pour PC, est lancé par Micropro au milieu de 1982. Il s'agit de la transcription à MS-DOS d'un programme de traitement de texte sous CP/M, un des tout premiers systèmes d'exploitation pour micro-ordinateur.

SSI se lance elle aussi dans la mise au point du traitement de texte sur PC. WordPerfect pour PC sera semblable à la version Data General. Toutefois sur un PC on n'a pas à se soucier d'avoir plusieurs utilisateurs simultanés. SSI introduisit à cette occasion les notes en bas de page et le correcteur d'orthographe.

WordPerfect pour PC sort le 18 novembre 1982. Il est adapté à l'imprimante Epson commercialisée par IBM. En 1983, Microsoft sort Word qui est généralement jugé inférieur à WordPerfect.

WordPerfect s'adapte aux machines MS-DOS non IBM (Victor 9000, DEC Rainbow, Tandy 2000, TI Professional etc.) et il est mis à jour pour tenir compte de l'arrivée de nouveaux périphériques (imprimantes, écrans, claviers, disques durs).

Le produit présente encore des défauts : certains pilotes d'imprimantes sont bogués et le manuel n'est pas jugé « professionnel ». Alors que les imprimantes « bêtes » accordaient la même largeur à chaque caractère, les nouvelles imprimantes « intelligentes » sont capables de calculer les intervalles convenables pour imprimer divers types de caractères. WordPerfect décide de placer les instructions pour imprimantes dans une table située hors du programme, ce qui permettra de l'adapter plus facilement aux nouvelles imprimantes. WordPerfect 3.0 sera ainsi capable de servir plus de cinquante types d'imprimantes.

SSI décide de mettre en place un centre d'appel gratuit pour les utilisateurs. Cette mesure fera beaucoup pour le succès commercial du produit. Le service sera renforcé au printemps 1990 par la création d'un « hold jockey », personne qui anime les appels en attente en diffusant de la musique, des commentaires etc.

Avec WordPerfect 4.0 en 1984 le manuel est amélioré, l'installation est plus simple, le dictionnaire plus riche ; cette version comporte les notes en

²⁶. WordPerfect comportera finalement des versions pour les *mainframes* d'IBM, les VAX, des machines Unix, l'Apple II, l'Amiga, l'Atari, le Macintosh, le PC etc.

fin de texte et le traitement des erreurs est amélioré. Wordstar est encore le leader mais Micropro scie la branche sur laquelle il était assis en lançant Wordstar 2000 qui, étant en rupture avec l'ergonomie antérieure, déconcerte ses clients les plus fidèles.

WordPerfect 4.1 sort à l'automne 1985. Il comporte la table des matières automatique, l'indexation automatique, la possibilité d'étaler les notes de bas de page sur des pages successives, la numérotation des paragraphes, le thesaurus, une vérification d'orthographe améliorée.

La part de marché de WordPerfect s'améliore car il tire profit des erreurs de ses concurrents : Wang a décidé d'ignorer le marché du PC et continue à produire des machines à écrire informatisées, ce qui entraînera sa chute. IBM a conquis grâce à la magie de son nom une bonne part de marché avec Displaywrite mais ce produit reste inférieur à WordPerfect²⁷.

Chez Micropro, le turn-over des programmeurs est élevé, ce qui empêche l'accumulation d'expertise. Lotus, leader sur le marché du tableur, comprend mal celui du traitement de texte. Microsoft est le seul concurrent dangereux, sa maîtrise du système d'exploitation lui donnant un avantage stratégique tant au plan commercial qu'au plan de la conception technique.

SSI prend en 1986 le nom de WordPerfect Corporation. L'arrivée de l'imprimante à laser et de l'interface graphique entraîne un changement des règles du jeu qui donne à Word l'occasion de rattraper WordPerfect. Il fallait en effet pour s'adapter aux nouvelles règles réécrire les parties du programme concernant les imprimantes et l'affichage à l'écran ; les mesures devaient être exprimées en centimètres et non plus en lignes et en espaces ; il fallait connaître les dimensions de chaque caractère dans chaque type et savoir charger les types sur les imprimantes. Alors qu'en mode texte l'ordinateur traitait un écran formé de 25 lignes de 80 caractères, soit 2000 boîtes, en mode graphique il travaillait avec 640 * 480 points, soit plus de 300 000 éléments : l'affichage à l'écran était plus lent.

Il était d'ailleurs difficile au plan stratégique de prévoir qui serait le vainqueur sur le marché de l'interface graphique pour PC : les concurrents étaient IBM avec TopView, Digital Research avec Gem et Microsoft avec Windows. Sortir une version de WordPerfect pour Windows aurait apporté un soutien à Microsoft qui, par la suite, pourrait être en mesure d'évincer WordPerfect.

WordPerfect, incapable de résoudre à la fois tous ces problèmes, se concentra d'abord sur l'imprimante à laser. WordPerfect 5.0 ne sera prêt qu'en mai 1988.

* *

En 1987 WordPerfect a 30 % du marché devant Micropro à 16 %, IBM à 13 % et Microsoft à 11 %. Il tire un argument commercial du besoin de compatibilité entre les divers documents produits par une entreprise, voire

27. Selon Peterson, alors que SSI voyait dans la programmation un art IBM la considérait comme une *industrie* et mesurait la production selon le nombre de lignes du code source. Il en résultait des programmes de qualité médiocre (Microsoft avait fait la même observation : voir note page 116).

par des entreprises différentes : les entreprises réclament un standard. La stratégie était de produire une version pour chaque plate-forme significative, puis d'intégrer WordPerfect avec les autres produits importants sur chaque plate-forme : sur le marché du PC avec le tableur Lotus 1-2-3 ; sur le marché du VAX avec All-in-One etc.

Pour ne pas dérouter les utilisateurs il fallait que WordPerfect 5.0 ne fût pas trop différent des versions précédentes. On pouvait cependant y introduire discrètement les possibilités graphiques du *desktop publishing* : pour intégrer texte et graphique sur un document, il faut savoir répartir le texte autour de boîtes contenant les graphiques, et savoir faire en sorte que selon les besoins ces boîtes restent en place ou au contraire se déplacent avec le texte dans le document. Les graphiques pouvant être volumineux, il fallait aussi savoir traiter de gros fichiers. MS-DOS offrant peu de possibilités, il fallait enfin écrire les logiciels nécessaires pour composer et publier des graphiques.

L'une des décisions les plus importantes fut de ne pas faire de 5.0 un produit pleinement Wysiwyg. Le Wysiwyg implique non seulement que les textes gras et soulignés, les fins de phrase et les fins de page s'affichent comme sur l'imprimé, mais aussi que les caractères soient des mêmes style et taille et que les notes de bas de page et les graphiques soient placés au bon endroit. Ce n'était pas facile avec MS-DOS qui ne fournissait pas beaucoup d'outils pour traiter l'écran. Windows, lui, fournissait les pilotes d'écran mais il était lent et peu fiable.

WordPerfect décida donc que l'écriture se ferait en mode texte, le mode graphique n'étant utilisé que pour visualiser la page à imprimer et y insérer les graphiques. Le Wysiwyg complet attendrait la version 6.0.

Les autres améliorations de la version 5.0 concernent la référence automatique (« voir p. 17 » se met à jour si le contenu de la page 17 est déplacé à une autre page), le support pour 1500 caractères y compris les caractères internationaux et diacritiques, l'impression intelligente (adapter au mieux le document à l'imprimante utilisée), la fusion de documents, les listes à puces, le suivi des modifications etc.

La diversité des matériels sur le marché contraignait WordPerfect à traiter des problèmes comme les suivants : « si l'utilisateur crée un document prévu pour l'imprimante à laser, puis l'emmène chez lui et tente de l'imprimer sur un autre type d'imprimante, WordPerfect devra-t-il reformater automatiquement le document ? Sinon, que se passera-t-il si le pilote de l'imprimante du bureau n'est pas présent sur la machine à domicile ? Si le document est reformaté automatiquement, l'utilisateur en sera-t-il informé, pourra-t-il annuler le reformatage ? »

En 1987, IBM et Microsoft lancent OS/2 et Presentation Manager, concurrent de Windows. En fait, Microsoft va miser sur Windows ; OS/2 et Presentation Manager deviendront des produits purement IBM. Craignant qu'un succès de Windows ne donne l'avantage à Microsoft, qui connaissant bien sa propre interface graphique pourrait mieux la maîtriser, WordPerfect va donner la préférence à OS/2.

Lorsque WordPerfect 5.0 pour MS-DOS sort enfin en mai 1988 il faudra surmonter des bogues dans le programme d'installation ainsi que dans certains pilotes d'imprimante et de carte graphique. Le produit a tout de même du succès : raccordé à une imprimante à laser, le PC fournit avec WordPerfect une qualité d'impression semblable à celle du Macintosh. WordPerfect prend par ailleurs une part du marché du *desktop publishing*.

* *

La version 5.1 sort à l'automne de 1989. Elle est encore sous MS-DOS mais comporte les menus déroulants et la souris. Elle apporte des améliorations dans le traitement des tableaux, la fusion de documents, la tabulation, ainsi qu'un jeu de caractères élargi. Son installation est plus facile qu'auparavant.

En 1990 Microsoft offre à WordPerfect Windows 3.0 en beta test, mais WordPerfect manque de programmeurs expérimentés en Windows et préférera aider au succès d'OS/2.

Cependant le 31 mai 1990 Microsoft sort Windows 3.0. *Alors les pires craintes de WordPerfect se réalisent* : beaucoup d'utilisateurs veulent Windows, même bogué, et à l'occasion ils prennent aussi Word. WordPerfect décide alors de laisser tomber OS/2 et renonce à sa version 5.2 pour MS-DOS afin de ne pas retarder la version Windows.

La mise au point de celle-ci est difficile. WordPerfect pour Windows ne sort qu'en novembre 1991. Le produit est lent et comporte des bogues ; néanmoins WordPerfect se vend toujours mieux que Word. Pour pouvoir programmer la version suivante, les programmeurs devront attendre de disposer du DDE (Dynamic Data Exchange) de Microsoft.

« WordPerfect for Windows » 6.0 sortira en 1993. WordPerfect prend alors 51 % du marché du traitement de texte pour Windows, mais Word est bien placé pour devenir le leader.

À la suite de l'échec d'une tentative d'entrée en bourse, WordPerfect sera vendue à Novell en 1994. Novell revendra en 1996 ses droits sur WordPerfect à Corel, qui en est aujourd'hui propriétaire.

WordPerfect a des millions d'utilisateurs mais c'est Microsoft qui, avec Word, domine aujourd'hui le marché du traitement de texte sur PC où il est concurrencé par quelques logiciels libres (notamment OpenOffice).

3.4.2 Histoire du tableur

Dans le langage des comptables américains, le mot *spreadsheet* désignait depuis toujours une grande feuille de papier divisée en lignes et en colonnes et utilisée pour présenter les comptes d'une entreprise. La meilleure traduction de ce mot en français est « tableau ».

En 1961 Richard Mattessich [127], professeur à Berkeley, développa en Fortran IV un *computerized spreadsheet* fonctionnant sur *mainframe*. Ce programme évaluait automatiquement certaines cases et permettait des simulations. Il est le précurseur des *electronic spreadsheets* d'aujourd'hui, que l'on appelle en français « tableurs ».

< 1961	Les comptables utilisent des tableaux sur papier .
1961	Computerized Spreadsheet de Mattessich en Fortran IV.
1978	Création de Software Arts.
1979	VisiCalc de Software Arts, commercialisé par VisiCorp, pour l'Apple II.
1981	VisiCalc est adapté à divers systèmes, notamment au PC d'IBM.
1982	Multiplan sous MS-DOS, de Microsoft.
1983	Lotus 1-2-3 , de Lotus.
1984	Excel pour le Macintosh, de Microsoft.
1985	Lotus achète Software Arts.
1987	Excel 2.0 pour le PC, de Microsoft.
1995	IBM achète Lotus. Excel est désormais leader sur le marché des tableurs.
> 2000	Gnumeric , KSpread , CALC etc. offerts en logiciel libre.

TAB. 3.1 – *Chronologie du tableur*

En 1978, Daniel Bricklin, étudiant à Harvard, devait établir des tableaux comptables pour une étude de cas sur Pepsi-Cola. Plutôt que de calculer à la main il préféra programmer « un tableau noir et une craie électroniques », selon sa propre expression. Son premier prototype, en Basic, pouvait manipuler un tableau de vingt lignes et cinq colonnes.

Bricklin se fit aider ensuite par Bob Frankston, du MIT. Celui-ci réécrivit le programme en assembleur et le condensa en 20 koctets pour qu'il puisse fonctionner sur un micro-ordinateur.

À l'automne 1978, Daniel Fylstra, ancien du MIT et rédacteur à *Byte Magazine*, perçut le potentiel commercial de ce produit. Il suggéra de l'adapter à l'Apple-II ainsi qu'aux systèmes HP85 et HP87.

En janvier 1979 Bricklin et Frankston créèrent Software Arts Corporation (Bricklin [27]); en mai 1979, la société Personal Software de Fylstra, nommée plus tard VisiCorp, lança la commercialisation de VisiCalc (cette appellation condense l'expression *Visible Calculator*).

VisiCalc était vendu 100 \$. Il avait déjà l'allure des tableurs d'aujourd'hui : les évolutions ultérieures les plus visibles porteront sur l'adjonction de possibilités graphiques ainsi que sur l'utilisation de la souris. Le succès ne fut pas immédiat mais néanmoins rapide. Dès juillet 1979 Ben Rosen publia une analyse prophétique (Rosen [177]).

Jusqu'alors seuls des hobbyistes, qui savaient programmer, pouvaient utiliser le micro-ordinateur : *VisiCalc était le premier logiciel qui permettait d'utiliser un ordinateur sans avoir à programmer*. Il contribua fortement à la pénétration du micro-ordinateur dans les entreprises. Des versions furent produites pour diverses plates-formes, notamment pour le PC d'IBM dès son lancement en 1981. Cependant les promoteurs de VisiCalc, empêtrés dans un conflit entre Software Arts et VisiCorp, ne surent pas faire évoluer leur produit assez rapidement.

Mitch Kapor avait travaillé pour Personal Software en 1980 et proposé un produit que les dirigeants de VisiCorp refusèrent parce qu'ils l'estimaient trop limité. Il créa Lotus Development Corporation en 1982 et lança Lotus 1-2-3 en 1983. Lotus 1-2-3 pouvait être adapté plus facilement que VisiCalc à divers systèmes d'exploitation et apportait des possibilités nouvelles : graphiques, bases de données, dénomination des cellules, macros. Il devint rapidement le nouveau tableur standard.

En 1985, Lotus achètera Software Arts et arrêtera la commercialisation de VisiCalc.

* *

Microsoft s'était intéressé au tableur dès 1980. En 1982, il lance Multiplan pour le PC. Ce produit n'aura pas grand succès aux États-Unis où Lotus 1-2-3 était dominant. Par contre il sera largement utilisé ailleurs et il ouvrira la voie aux autres applications produites par Microsoft.

En 1984, Microsoft sort Excel pour le Macintosh. Le produit tire parti de l'interface graphique, des menus déroulants, de la souris : tout cela le rend plus commode que Multiplan. Tout comme VisiCalc avait contribué au succès du PC, Excel contribuera au succès du Macintosh.

En 1987 Microsoft sort Excel pour le PC : ce sera l'application phare de Windows. La principale amélioration par rapport à Lotus 1-2-3 est la possibilité de programmer de véritables applications avec des macro-instructions (l'utilisateur individuel ne s'en servira cependant pas beaucoup).

En 1987, Microsoft Works inaugure la famille des *office suites* en offrant dans un même assemblage le tableur, le traitement de texte et le logiciel graphique. Excel sera jusqu'en 1992 le seul tableur disponible sous Windows.

À la fin des années 1980, Lotus et Microsoft dominent le marché malgré l'arrivée de plusieurs autres tableurs (Quattro de Borland, SuperCalc de Computer Associates etc.). La concurrence est vive et suscite des batailles juridiques : procès entre Lotus et Software Arts, gagné par Lotus en 1993 ; procès entre Lotus et Mosaic d'une part, Paperback de l'autre, gagnés par Lotus en 1987. Lotus gagnera toutes ses batailles juridiques, mais perdra contre Microsoft la bataille pour la domination du marché.

En 1990, un juge lèvera le copyright de Lotus sur l'interface utilisateur, estimant que « rien dans cette interface n'est inséparable de l'idée du tableur ». En 1995, IBM achète Lotus alors qu'Excel domine le marché.

Plus de 20 tableurs sont aujourd'hui offerts dans le monde du logiciel libre. Gnumeric est souvent distribué en même temps que Linux. Parmi les autres tableurs, les plus connus sont KSpread et CALC.

* *

Si VisiCalc présente dès le début l'aspect qui nous est familier, le tableur s'est progressivement enrichi. L'adresse des cellules, d'abord notée selon le format L1C1 (R1C1 pour les anglophones) a pu ensuite s'écrire sous la notation condensée A1. L'existence de deux types d'adresse (adresses relatives, adresses absolues de type \$A\$1) a allégé la programmation. L'introduction des feuilles et des liens a permis de doter le tableur d'une troisième di-

mension (la feuille s'ajoutant à la ligne et à la colonne), voire d'un nombre quelconque de dimensions si l'on relie plusieurs tableurs.

Lotus 1-2-3 a apporté les outils graphiques qui facilitent la visualisation des résultats. Les macros (également introduites par Lotus 1-2-3 en 1983, puis perfectionnées par Microsoft) permettent de programmer des applications sur le tableur. La souris (à partir de 1984 avec Excel sur le Macintosh) a facilité la sélection des plages de cellules et la dissémination des formules par glissement du pointeur. Le solveur (introduit en 1990 par Frontline) permet de résoudre des problèmes de calcul numérique, d'économétrie, de recherche opérationnelle etc.

* *

Dans l'entreprise, le tableur est utilisé pour les simulations, les calculs sur les séries chronologiques, la comptabilité, la préparation de rapports ou de déclarations fiscales. Des fonctions simples sont utilisées de façon répétitive pour faire des additions et calculer des moyennes. Les utilisations scientifiques (calcul numérique, visualisation de statistiques, résolution d'équations différentielles) sont plus compliquées et moins répétitives.

Pour l'utilisateur de base, le tableur n'est que la fusion électronique du papier, du crayon et de la calculette. Il n'a généralement pas été formé à s'en servir et il est le plus souvent peu conscient des conséquences que risque d'avoir une erreur. C'est un expert dans son métier et il ne se considère pas comme un programmeur. Il veut traiter rapidement son problème et ne souhaite ni recevoir les conseils d'un informaticien, ni partager son expertise avec lui. Il est en pratique impossible de lui imposer des méthodes strictes de programmation ou de vérification. Son développement progressera par essais et erreurs : il construit un premier prototype puis le modifie jusqu'à ce qu'il réponde à ses besoins. Pressé d'arriver à ses fins, il néglige de documenter le programme. Celui-ci ne pourra pratiquement jamais être réutilisé par quelqu'un d'autre et son créateur lui-même aura du mal à le faire évoluer ou à le maintenir.

Les dirigeants de l'entreprise, pour leur part, n'utilisent généralement pas le tableur mais sont destinataires de tableaux de bord et autres *reportings* imprimés sur papier mais construits sur des tableurs. Ils lisent ces tableaux comme s'ils provenaient d'un traitement de texte et sans concevoir les calculs dont ils résultent.

L'entreprise donne donc au tableur un rôle ambigu : c'est un outil de travail commode mis à la disposition de tous, mais générateur d'erreurs et dont les programmes sont difficiles à entretenir. Vu l'importance prise par le tableur dans le système d'information de l'entreprise, il est utile de repérer les erreurs et d'utiliser les méthodes qui permettent de les éviter (O'Beirne [148]).

* *

Une enquête a montré que la majorité des tableurs contenaient des cellules fausses et qu'en moyenne 3 % des cellules d'un tableur étaient erronées (Clermont [38] et Panko [150]). Certaines erreurs ont eu des conséquences graves :

1) Les données utilisées pour passer une commande sont désuètes : 30 000 pièces à 4 \$ sont commandées au lieu de 1 500, ce qui entraîne une perte de 114 000 \$.

2) Dans une étude prévisionnelle, les sommes en dollars sont arrondies à l'unité : le multiplicateur de 1,06 qui représente l'effet de l'inflation est alors arrondi à 1 et le marché d'un produit nouveau est sous-estimé de 36 millions de dollars.

3) Un tableur a été programmé par une personne qui a quitté l'entreprise et n'a pas laissé de documentation : le taux d'actualisation utilisé pour calculer la valeur actuelle nette des projets est resté à 8 % entre 1973 et 1981 alors qu'il aurait dû être porté à 20 %, d'où des erreurs dans le choix des investissements.

4) Dans la réponse à un appel d'offre une addition est inexacte (des rubriques ajoutées à la liste n'ont pas été prises en compte) : l'entreprise sous-estime de 250 000 \$ le coût du projet. Elle intentera un procès à Lotus.

5) Un comptable fait une erreur de signe lors de la saisie d'un compte de 1,2 milliards de dollars : l'entreprise prévoit un profit de 2,3 milliards et annonce une distribution de dividendes. Finalement elle constate une perte de 100 millions de dollars et doit déclarer qu'aucun dividende ne sera distribué, ce qui dégrade son image auprès des actionnaires.

6) En 1992, 10 % des tableurs envoyés aux inspecteurs des impôts britanniques pour le calcul de la TVA contenaient des erreurs. Il en est résulté une perte de recettes de cinq millions de livres sterling.

Certaines des erreurs relevées dans l'utilisation du tableur sont de celles que l'on peut rencontrer dans d'autres démarches : la représentation du monde réel par un modèle peut être non pertinente ou dégradée par des défauts du raisonnement (additionner des données hétéroclites, des ratios etc.). D'autres erreurs sont commises lors de la programmation : on confond référence relative et référence absolue, on se trompe dans la syntaxe des formules (notamment dans l'utilisation des parenthèses). Ces erreurs, qui révèlent une mauvaise compréhension du fonctionnement du tableur, sont ensuite disséminées par la réplique des cellules. Presque toujours enfin on néglige de documenter le tableur, ce qui rendra sa maintenance difficile surtout si l'on a programmé des macros.

Puis viennent les erreurs commises lors de l'utilisation : erreurs de saisie, erreur dans la correction d'une formule, remplacement *ad hoc* d'une formule par une constante qui, restant dans le tableau, polluera les calculs ultérieurs, mauvaise définition de l'aire couverte par une formule, absence de mise à jour de cette aire lorsque des lignes sont ajoutées.

Utiliser les macros comme des boîtes noires (par exemple pour le calcul du taux de rentabilité d'un projet) peut interdire de traiter convenablement le cas particulier que l'on étudie. Il arrive aussi que le solveur converge mal : le prendre au pied de la lettre donne un résultat aberrant.

L'édition des tableaux sur papier est l'occasion d'erreurs de présentation : tableaux sans titre, sans intitulé de ligne et de colonne, sans nom d'auteur, sans date ni indication de la période représentée ; erreurs sur les unités de mesure (euro à la place de dollar, millions à la place de milliards).

On relève enfin des erreurs dans les graphiques : représenter une série chronologique par un histogramme, ou pis par un fromage, au lieu d'une courbe ; utiliser une courbe au lieu d'un histogramme pour une distribution ; quand on utilise conjointement deux échelles, mal représenter les évolutions relatives etc.

3.4.3 Leçons de l'histoire de la bureautique

Pour étudier l'histoire du tableur ou du traitement de texte, il faut dépouiller beaucoup de documents. Deux faits sautent alors aux yeux : l'un concerne la documentation elle-même, l'autre concerne les inventeurs.

La plupart des documents accordent une grande place à l'identité des innovateurs et à l'évolution du marché : *qui* a inventé le produit, et quand ; *quand*, et avec qui, il a créé son entreprise ; *quand*, et comment, s'est manifestée la concurrence ; *comment* ont évolué les parts de marché ; *comment* se sont réglés les conflits. Les indications sur la nature du produit partent presque toutes du point de vue de l'utilisateur, car il s'agit d'expliquer un succès commercial. Si l'idée qui a guidé la conception est indiquée, on ne trouve pas grand-chose sur les choix fonctionnels ou d'architecture, ni sur les compromis auxquels le réalisateur a été contraint : le passage de l'idée au produit s'est déroulé dans une boîte noire que la documentation entrouvre à peine.

Il est vrai qu'il est difficile de décrire ce passage, qui laisse le plus souvent peu de traces documentaires. Cela ne donne que plus de valeur aux ouvrages qui, comme celui de Tracy Kidder [101] (voir page 48), décrivent en détail le cheminement d'une réalisation technique. Dans la plupart des cas, l'inventeur est récemment sorti de l'université, ou même encore un étudiant. Il est au courant des techniques récentes et, comme il n'a aucune expérience, son esprit est libre de préjugés. Il imagine le nouveau produit en partant de ses propres besoins. Beaucoup des produits ainsi conçus n'aboutissent à rien de durable, ceux-là seront oubliés : l'histoire ne garde en mémoire que les produits réussis. Notre jeune diplômé, si son produit rencontre le succès, devient un entrepreneur : il découvre les soucis du recrutement, de la commercialisation, des partenariats, de la gestion, du financement. Pour défendre son territoire il s'englué dans des procédures. Il perd ainsi sa capacité à innover et d'ailleurs son savoir technique, naguère pointu, devient banal. Il laisse passer des occasions qu'il ne perçoit plus et se fait doubler par d'autres personnes tout juste sorties de l'université. Bientôt la survie de son entreprise est menacée...

Le système est innovant parce que le flux de jeunes innovateurs ne tarit pas. Par contre si l'on considère les individus, le système paraît stérilisant : tout se passe comme si le fait de mettre une idée sur le marché inhibait la possibilité d'en avoir une deuxième. Quant aux entreprises, elles naissent, croissent et meurent ou se font acheter en quelques années. Microsoft est l'exception qui confirme la règle car si Bill Gates innove c'est en entrepreneur, en organisateur plus qu'en technicien : il reprend les produits inventés par d'autres et les articule en système, avec un sens aigu des économies d'envergure et du fonctionnement du marché.

3.5 Esquisse de prospective

Rappelons ce qu'était l'informatique dans les entreprises en 1988. La Toile n'existait pas (elle a été inventée en 1991) et on ne connaissait guère l'Internet. L'informatique ne traitait pratiquement que des données structurées. La messagerie, les workflows, la documentation électronique existaient sur de gros systèmes, mais seuls certains informaticiens les utilisaient car ils étaient peu ergonomiques. Les réseaux locaux étaient rares (ils ne se multiplieront qu'à partir de 1989). Le téléphone mobile était lourd (il n'équipait pratiquement que des automobiles) et coûteux.

Les micro-ordinateurs étaient chers : un PC à 16 MHz, avec un disque dur de 80 Mo et une RAM de 2 Mo, coûtait plus de 10 000 € aux prix de 2005. L'ergonomie était rudimentaire à l'exception de celle des Macintosh (Windows 95 date, comme son nom l'indique, de 1995).

Qu'avons-nous acquis depuis 1988 ? L'informatique de communication s'est développée : la messagerie est entrée dans les mœurs ainsi que la documentation électronique (Intranet et Toile). L'ordinateur a acquis l'ubiquité : la machine dont chacun dispose permet d'accéder *via* le réseau à des ressources informatiques (mémoire, puissance de traitement) dont la localisation physique est indifférente. Le travail assisté par ordinateur s'est généralisé : l'automate soulage le travail mental de l'être humain en l'assistanant dans les tâches de classement, recherche, transcription et calcul.

Le système d'information définit le langage de l'entreprise dont il incorpore la sémantique. Il outille son action, ses processus de production. Il équipe progressivement tout le personnel. La maîtrise intellectuelle et pratique du système d'information suppose que l'on sache urbaniser sa structure d'ensemble et modéliser chacun des processus. L'automatisation (partielle) des processus doit être associée à la vigilance qui permettra à l'agent opérationnel de trouver une solution raisonnable quand il rencontre un cas non prévu lors de la programmation de l'automate.

* *

Plaçons nous par anticipation en 2015. Les composants essentiels de l'informatique communicante existent déjà aujourd'hui (processeurs, mémoires, réseaux) ; le changement à cette échéance résidera donc moins dans la nouveauté des composants (dont toutefois les performances auront depuis 2005 été, selon la loi de Moore, multipliées par $2^{10/1,5} \approx 100$) que dans la transformation des interfaces et protocoles permettant de les commander et de les faire communiquer, donc dans une évolution des conditions de leur utilisation.

En 2015, si l'on raisonne par extrapolation tendancielle, téléphone portable et PC se seront miniaturisés au point d'avoir fusionné et d'avoir absorbé le GPS, la caméra, le magnétophone et divers capteurs (électrocardiogramme, tensiométrie etc.). Les réseaux permettront une connexion permanente à haut débit selon le protocole TCP/IP (voir page 340), y compris pendant les déplacements. Les personnes, les objets seront ainsi équipés de ressources informatiques et de communication intégrées dans des composants minuscules. Les thèmes principaux de l'offre porteront peut-être

les noms suivants : « communicateurs personnels », « containers d'information », « télédiagnostic des équipements » etc. De nombreux « téléservices » pourront les utiliser.

L'ordinateur *wearable*, portable au sens où l'on dit que l'on « porte » des vêtements, sera devenu discret et pratique. Après la révolution du téléphone mobile, ce sera celle, bien plus radicale, de l'ordinateur mobile. Les personnes disposeront de fonctions informatiques et de communication qui équiperont non plus le bureau, mais le *corps* lui-même ; la chaleur et les mouvements du corps fourniront l'énergie nécessaire. L'écran sera incorporé aux lunettes, ou bien il s'affichera par exemple sur une feuille que l'on pourra déplier comme un journal. Processeur, mémoire et disque dur seront intégrés dans un boîtier qui servira de *palmtop* et de clavier. Le son sera fourni par un balladeur ou par un composant inséré dans l'oreille (l'oreillette Bluetooth en est un précurseur). Les commandes seront saisies par reconnaissance vocale ou par clavier. Les diverses parties de l'équipement communiqueront par câble ou ondes à courte portée.

* *

Alors l'utilisateur peut consulter des ressources, recevoir des alarmes etc. : son équipement reçoit et envoie des messages écrits et vocaux et assiste sa mémoire, dans la continuité des services que rend aujourd'hui l'agenda sur papier, en l'enrichissant par l'accès à des ressources encyclopédiques et des moteurs de recherche.

Ainsi *l'ubiquité logique* (disponibilité des données et des outils de traitement quel que soit l'endroit où l'on se trouve) est absolue, sous la seule réserve du contrôle d'accès. L'ubiquité physique reste, elle, limitée : si notre voix et notre image peuvent se trouver simultanément en divers points du monde, il n'en est pas de même de notre corps qui peut toutefois être représenté sur le réseau par une image « virtuelle » à trois dimensions (Quéau [169]).

Les personnes qui veulent communiquer avec l'utilisateur peuvent lui être présentées par leur « carte de visite » comportant une photographie (enrichissement de l'identification d'appel) et l'utilisateur a le choix entre communication synchrone et asynchrone.

La personne équipée qui se déplace dans un environnement d'objets communicants reçoit les signaux émis par ces objets et les interprète. Elle peut aussi recevoir les signaux émis par les équipements des autres personnes (identifier amis et relations dans une foule etc.).

Les objets eux-mêmes sont munis de ressources informatiques qui facilitent la « traçabilité » des biens de consommation (origine, composition chimique et fraîcheur des produits alimentaires, identification des fournisseurs ayant participé à l'élaboration d'un produit composite etc.) La traçabilité des produits, notamment alimentaires, constitue un avantage compétitif qui se traduit soit par l'acceptation de prix plus élevés, soit par l'élimination progressive des produits non tracés. Des étiquettes rayonnantes permettent de les identifier, puis de trouver sur la Toile les informations complémentaires.

L'appartement est truffé d'objets communicants aux fonctions diverses. Il est équipé d'un ordinateur central relié au monde par des accès à haut débit (les accès ADSL et les paraboles pour satellites en sont une préfiguration) qui constitue le centre du réseau des objets communicants, organise les fonctions informatiques, audiovisuelles et télécoms du ménage, pilote le chauffage, l'éclairage, l'arrosage du jardin etc. selon les consignes fournies par l'utilisateur.

Le terme « ordinateur » ne désigne plus ici une machine, mais un ensemble de fonctions résidant sur des machines diverses, y compris sur des machines situées hors de l'appartement mais fonctionnant sous le contrôle du ménage. L'utilisateur ne dispose plus comme aujourd'hui de plusieurs ordinateurs (un au bureau, un au domicile, un *palmtop*, un ordinateur portable, et en outre un téléphone filaire et un téléphone mobile) entre lesquels il doit recopier les données : il dispose d'une *ressource informatique* unique (donc essentiellement cohérente), localisée sur des serveurs dont l'emplacement géographique lui indiffère et sur lesquels traitements et données se répartissent. Il accède à cette ressource par des interfaces diverses sans que cela altère l'unité de celle-ci : son ordinateur mobile, mis en face d'un écran-clavier confortable, active celui-ci à volonté. À la fois informatique et téléphonique, cette ressource gère les messageries écrites et vocales, garde mémoire des communications vocales ou des conversations. Elle est connectée en permanence au Web sur lequel elle réalise des missions de recherche et de tri.

* *

Dans ce scénario, *rien ne dépasse les possibilités de la technique actuelle* qui sont plus importantes qu'on ne le croit communément²⁸. Ce qui est nouveau, c'est l'intégration des applications qui permet de supprimer les ressaisies ; c'est aussi un filtrage sélectif permettant de trier sur la Toile l'utile de l'accessoire. L'utilisateur peut ainsi être assisté ou éclairé dans chacune de ses actions : la logique de *l'assisté par ordinateur* déploie ses implications. L'offre est commode, pratique, une haute complexité technique étant masquée par la facilité de l'usage.

Cette évolution, d'ores et déjà en préparation dans les centres de recherche, chez les fournisseurs et les opérateurs télécoms, comporte des risques évidents :

1) *risque de dépendance* envers un système qui assisterait l'utilisateur en permanence : un nouveau savoir-vivre, une nouvelle hygiène, sont ici nécessaires. De même qu'il est aujourd'hui déconseillé de regarder la télévision sans discontinuer, il sera déconseillé demain d'utiliser en permanence l'automate.

Un système qui permet de recevoir en temps réel alarmes, messages et communications, qui permet à d'autres personnes de vous localiser, qui peut à tout moment accéder à des ressources (images, données, textes, sons,

²⁸ L'ensemble des conversations auxquelles participe un même individu durant sa vie, des cours qu'il suit etc. peut une fois transcrit en mode caractère être horodaté, indexé et stocké sur un CD.

jeux), peut en effet se révéler oppressant. Il doit comporter divers niveaux de veille, de l'arrêt total à l'ouverture totale, en passant par le blocage sélectif de certaines communications. Il faudra savoir se débrancher et utiliser la communication asynchrone : déjà, aujourd'hui, il faut dans certains lieux savoir débrancher son téléphone mobile.

2) *sécurité* : l'importance prise par l'automatisation a pour corollaire l'obligation de *contrôler les automates*, car personne ne peut faire entièrement confiance à des automatismes pour sa vie courante. La communication entre automates, les actions qu'ils déclenchent doivent pouvoir être traçables et contrôlables ; l'utilisateur doit disposer d'interfaces commodes pour les paramétrer ; elles doivent être sécurisées pour éviter les fausses manœuvres ; la réplication des données sur plusieurs serveurs est nécessaire pour limiter le risque de perte.

La protection de la vie privée suppose enfin que l'automate soit protégé contre toute tentative d'indiscrétion.

3) *risque de rejet* : le monde dans lequel vivra l'utilisateur est différent de celui que nous connaissons aujourd'hui : les appareils seront plus discrets mais leurs fonctionnalités seront omniprésentes. Elles susciteront des réactions de rejet comme l'ont fait en d'autres temps le téléphone, l'ordinateur, le Minitel et les équipements électroménagers (machine à laver, aspirateur etc.)

Il faut être averti de ces risques mais il ne faut pas les exagérer : parmi les périls auxquels l'humanité est confrontée, ils ne sont certainement pas les plus graves (voir la note page 27) et ils doivent être mis en balance avec les avantages qu'apporte l'informatique.

* *

La plupart des produits offerts au consommateur seront des assemblages auxquels plusieurs entreprises auront contribué dans le cadre d'accords de partenariat. Soyons optimistes : les entreprises se seront dotées de compétences en ingénierie d'affaires et il leur sera devenu plus facile d'assurer l'interopérabilité entre leurs systèmes d'information. Les systèmes d'information seront mieux maîtrisés, qu'il s'agisse d'urbanisation ou de modélisation : les entreprises auront surmonté les difficultés sociologiques et intellectuelles qu'elles rencontrent aujourd'hui, et disposeront de méthodes et outils de modélisation. On ne parlera plus d'applications, mais de processus et de composants. Les problèmes de normalisation auront été surmontés...

Deuxième partie

Le côté de l'entreprise

Chapitre 4

Qu'est-ce qu'une *entreprise* ?

Pierre Bourdieu, citant Heidegger, lui a fait dire « Pour celui qui porte des lunettes - objet qui pourtant, selon la distance, lui est proche au point de lui “tomber sur le nez” - cet outil est, au sein du monde ambiant, plus éloigné de lui que le tableau accroché au mur opposé » (Bourdieu [22]). Nous ne voyons pas l'air dans lequel nous baignons et qui est nécessaire à notre survie : si une pièce ne contient pas de meubles, nous dirons qu'elle est vide alors qu'elle contient des dizaines de kilogrammes d'air¹. Les êtres humains ont vécu pendant des millions d'années sans rien savoir de la circulation du sang ni des mécanismes de la digestion ; nous pensons sans savoir comment notre cerveau fonctionne. Nous ne voyons pas le milieu qui nous baigne et les artefacts qui nous sont proches nous paraissent naturels. Tout se passe comme si le lait et l'huile étaient produits par l'épicerie, comme si la lumière était produite par une pression sur l'interrupteur. Il faut une panne, une crise, pour que leur origine nous revienne à l'esprit.

* *

« Je ne sais pas ce que c'est qu'un système d'information », vous dit-on. Si cet interlocuteur ne voit pas le système d'information, c'est parce qu'il est y immergé. Il ne pourra le voir qu'à l'occasion d'une panne ou au prix d'un effort de réflexion. Citez-lui des exemples : téléphoner, c'est utiliser un système d'information qui interprète le numéro que vous avez composé, établit le circuit de communication et code le signal vocal pour le transporter. Retirer des billets de banque au distributeur automatique, c'est utiliser le système d'information qui authentifie votre carte et débite votre compte. Lire des messages, leur répondre, c'est utiliser le système d'information qui transporte et stocke les messages dans votre boîte aux lettres électronique en l'attente de leur consultation. Naviguer sur la Toile, c'est encore utiliser un système d'information.

1. La masse volumique de l'air à 17°C, au niveau de la mer, est de 1,2 kg/m³.

Dans l'entreprise, vous utilisez les « applications » qui correspondent à vos fonctions. Êtes-vous conseiller dans une agence bancaire? le système d'information vous permet de consulter les comptes des clients, de réaliser les opérations qu'ils vous demandent. Êtes-vous ouvrier en mécanique générale? le système d'information fournit les cotes de la pièce à fabriquer et aide à programmer votre machine. Êtes-vous un gestionnaire ou, comme on dit, un *manager*? c'est dans le système d'information que vous trouverez les indicateurs, les alarmes nécessaires à votre mission de supervision.

Je sais bien tout cela, répond l'interlocuteur rétif; mais vous ne me dites toujours pas ce qu'est un système d'information. Il ne fournit en effet que des *données*. Quel est le traitement à leur appliquer pour qu'elles deviennent des *informations*? Et comment peut-on passer de l'information à la *connaissance*?

Pour lui répondre selon les règles de la logique il faudrait définir les termes qu'il utilise mais la conversation s'égarerait. Raisonner par analogie est moins rigoureux mais permet de garder les pieds par terre. « Ces mêmes questions se posent, répondrez-vous, à propos d'activités aussi quotidiennes que la lecture ou la conversation. Comment faites-vous pour conférer un sens à un texte? Pour comprendre ce que vous dit quelqu'un? Pour tirer parti des connaissances ainsi acquises? Il faut avoir répondu à ces questions-là avant de se les poser à propos du système d'information: alors elles s'éclairent d'elles-mêmes. »

Mais sans doute votre interlocuteur ne s'y intéressera pas, de même qu'il ne s'intéresse pas à la façon dont son appareil digestif fonctionne: cela marche tout seul, pourquoi s'en soucier? Il préférera donc poser encore une autre question. « Que cherche-t-on à faire lorsqu'on met en place un système d'information? S'agit-il de clarifier la forêt de l'entreprise ou de changer sa stratégie? »

Le mieux est de maintenir la discussion au ras du sol en recourant de nouveau à l'analogie: « Pourquoi lisez-vous des livres, des journaux? Pourquoi avez-vous appris à parler? À quoi vous sert le langage? Vous faites effort, sans doute, pour éviter la confusion et les incohérences. Vous souhaitez que votre mémoire garde trace des choses importantes et qu'elle oublie les autres. Mais comment distinguez-vous les choses importantes? Ne vous référez-vous pas, pour les trier, à ce que vous voulez *faire*? Wittgenstein disait: "Don't ask for meaning, ask for use", « Ne te demande pas ce que ça veut dire, demande toi à quoi ça sert ».

« Si vous savez répondre à ces questions en tant que personne, vous saurez aussi leur répondre en ce qui concerne l'entreprise et son système d'information: il s'agit en effet de la même chose, mais transposée à une institution. Cette transposition fait apparaître la complexité de démarches qui, dans la vie quotidienne, nous sont si habituelles que nous les croyons simples. » L'interlocuteur vous dira alors peut-être piteusement « Je ne sais pas ce que c'est qu'une entreprise ». Rassurez-le: il n'est pas le seul dans son cas. Comme l'entreprise nous crève les yeux, elle est invisible. C'est cela qui rend invisible le système d'information, qui n'est rien d'autre que le langage de l'entreprise.

4.1 Le point de vue du système d'information

Lorsqu'on considère le système d'information d'une entreprise, on adopte sur celle-ci un point de vue particulier : celui de son *fonctionnement* et donc de son *efficacité*. Mais avant de parler d'efficacité il faut connaître le *but* que l'action vise, et qu'il s'agit d'atteindre au moindre effort.

Quel est donc le but de l'entreprise ? Beaucoup de personnes disent que c'est de « faire du profit », de « produire de l'argent ». On peut leur objecter que pour atteindre ce but-là, la prostitution ou la prédation des patrimoines mal protégés sont plus rapides et moins fatigantes que de créer, de gérer une entreprise. Certes, répondent-ils, mais il s'agit de s'enrichir par des moyens légaux et honorables.

Mais si l'honneur est en jeu, cela ne signifie-t-il pas que le but est autre, en fait, que de « produire de l'argent » ?

* *

Faisons le tour des représentations courantes de l'entreprise. L'économiste dit qu'elle a pour but de maximiser le profit : cela lui permet de recourir aux mathématiques une fois le profit défini comme fonction d'autres variables. Le dirigeant, reproduisant le système d'autorité de l'Église et de l'armée, y voit une structure hiérarchique². Le financier la considère comme une entité fiduciaire : elle doit susciter la confiance, avoir du *crédit*. Elle est le lieu de la carrière du cadre, auquel elle offre une échelle qu'il s'efforce de gravir, un terrain de compétition, le socle de son identité sociale (« cadre supérieur à Air France », « directeur chez Alcatel », « associé chez McKinsey », « ingénieur chez Areva » etc.).

Le salarié non cadre y voit la « boîte » qui en échange d'une partie de son temps lui procure, avec le salaire, les ressources nécessaires à sa vie matérielle. Le syndicaliste la perçoit comme un terrain de lutte : selon sa tendance, il défendra le salarié non cadre, le cadre, ou le syndicat lui-même qui ambitionne parfois de cogérer l'entreprise, voire de la diriger. L'homme du marketing la perçoit comme une « marque » capable de séduire et fidéliser la clientèle. L'ingénieur pense qu'elle produit des biens et services à partir des consommations intermédiaires, de la main d'œuvre, des techniques qu'incorporent les machines. L'informaticien, qu'elle utilise les ordinateurs, réseaux et logiciels dont il est le maître. Le comptable, qu'elle émet et reçoit des effets de commerce qu'il classe pour évaluer, conformément aux règles admises, les flux qui s'accumulent dans le bilan.

Le dirigeant, qui incarne la légitimité, doit composer avec d'autres pouvoirs : chaque directeur s'approprie une « plate-bande » qui, en descendant l'arbre hiérarchique, se subdivise en fiefs d'ampleur décroissante mais tous bien gardés ; des réseaux, tissés autour des écoles d'ingénieurs, syndicats ou partis politiques, se confortent à l'occasion par une corruption discrète mais habituelle.

2. « Hiérarchie » signifie étymologiquement « autorité sacrée », de *ἱερός* (sacré) et *ἀρχή* (commandement). Ce mot a désigné d'abord le pouvoir de l'évêque sur son diocèse.

Aux pouvoirs internes s'ajoutent des pouvoirs externes : le conseil d'administration, qui nomme le dirigeant et peut le révoquer *ad nutum* ; le pouvoir politique (gouvernement, élus) et administratif (préfecture, « Bruxelles », direction du Trésor, impôts, sécurité sociale, direction du travail etc.) ; le banquier qui propose, accorde ou retire des liquidités dont il fixe le prix ; l'actionnaire, nerveux comme un cheval ombrageux, qui détermine le cours de l'action ; les amateurs d'OPA à l'affût d'une baisse du cours. Pouvoirs internes et externes communiquent : les réseaux et les directeurs sont en relation avec les pouvoirs politique et administratif.

Étant tout cela à la fois, l'entreprise ne peut se réduire à une définition, même si la trivialité du *business is business*, le « sérieux » des ingénieurs, l'« autorité » des dirigeants, prétendent chacun la résumer en quelques phrases. C'est une entité organique, historique, culturelle, sociologique, le théâtre d'une *Comédie humaine* qui n'a pas encore rencontré son Balzac : la littérature ne rend pas compte aujourd'hui de la vie de l'entreprise, si l'on excepte quelques caricatures et romans policiers³. Il est surprenant que la production symbolique, qui prépare l'imaginaire à interpréter l'expérience, ne se soit pas encore sérieusement intéressée à ce lieu où chacun passe l'essentiel de ses journées et enracine, avec son identité sociale, les projets, angoisses et désirs qui entourent celle-ci.

* *

Si l'on aborde l'entreprise sous l'angle du système d'information, on adopte un tout autre point de vue que ceux que nous venons d'énumérer : il s'agit en effet de définir ce que doit faire l'automate pour assister les opérations qu'elle réalise.

La première question que l'on doit se poser est alors : « que *produit* cette entreprise ? », qu'il s'agisse de biens, de services ou, comme c'est le cas de plus en plus souvent, d'une combinaison des deux ; qu'il s'agisse de produits finals⁴ destinés aux clients, ou de produits intermédiaires, de « services support » qui alimentent ou favorisent la production.

Il n'est pas toujours facile de définir ce que produit une entreprise ou l'une de ses directions. Que produisent par exemple une direction des achats ? une direction financière ? un grand magasin ? une banque ? Mais si l'entreprise ne sait pas dire, ne sait pas se représenter ce qu'elle produit, elle aura du mal à s'organiser et sera tentée par la routine. La première question que suscite le système d'information est donc salubre⁵.

La deuxième question l'est aussi : « sachant ce que l'entreprise produit, comment s'y prend-elle pour produire ? Comment est bâti son *processus* de production ? »

3. Pilhes [161], Grisham [74], Crichton [40] et [41].

4. En français classique (cf. le *Dictionnaire de la langue française* de Littré), le masculin pluriel de « final » est « finals » et non « finaux » : on sourit lorsqu'on entend quelque chose qui sonne comme « les utilisateurs finaux »... Le *Larousse* tolère cependant cette tournure.

5. Pour la direction des achats, voici la réponse de Christophe Berthier : elle produit des contrats avec les fournisseurs. C'est évident, mais il fallait y penser.

Un dirigeant d'une grande banque m'a dit un jour que dans son entreprise le mot « processus » était jugé abstrait, théorique, loin du réel ; quant au mot *workflow*, on le trouvait incompréhensible. Que certaines personnes pensent que le processus de production est quelque chose d'abstrait, c'est un fait sans doute mais il est consternant. Quoi de plus authentiquement *concret*, en effet, que les *produits* de l'entreprise et que la façon dont on les produit, c'est-à-dire le processus de production ? Quant au *workflow*, ce n'est rien d'autre que l'informatisation du processus...

Ce qui est concret pour ces personnes, ce sont sans doute les habitudes de travail, l'unité à laquelle elles sont affectées, le chef, les collègues et subordonnés, le grade et la carrière, le bureau, les couloirs et la cantine. Le produit de l'entreprise, le processus de production leur paraissent abstraits tout comme le sillon est abstrait pour le bœuf qui tire fidèlement la charrue sans concevoir à quoi cela peut servir. Que le changement soit alors si difficile, cela ne doit pas surprendre : il faut un effort surhumain pour se sortir de l'ornière habituelle, après quoi on retombe lourdement dans une nouvelle ornière.

Il est étonnant que l'on demande encore aux agents opérationnels d'avoir les qualités du bœuf de labour alors que l'on dit attendre d'eux qu'ils soient capables d'initiative, qu'ils sachent interpréter les situations et prendre des responsabilités. Si l'on pense vraiment (et non seulement en paroles) que l'entreprise doit désormais être décentralisée, réactive, souple, évolutive etc., il faut pourtant qu'ils puissent savoir à quoi sert ce qu'ils font et se représenter le processus dans lequel ils interviennent.

La difficulté à percevoir les choses sous l'angle du processus a peut-être une autre cause, plus profonde. Héritiers de la philosophie grecque, nous sommes habitués à raisonner par concepts. Notre pensée est à l'aise devant les choses qui restent conformes à leur définition, à leur *forme*, mais mal à l'aise quand il faut concevoir une *transformation*. Or un processus est l'itinéraire d'une transformation qui va de la matière première au produit fini ; dans les systèmes d'information, cela se concrétise par le « cycle de vie des objets », qu'il est si difficile de modéliser (Shlaer et Mellor [131]). La pensée chinoise, qui conçoit l'évolution plus volontiers que la stabilité (Jullien [93]), ne serait-elle pas la plus agile quand il faut concevoir un processus ?

* *

Quand on a balisé les processus de production (et peut-être, à cette occasion, repéré des défauts à corriger dans les procédures habituelles) il faut encore définir les êtres à identifier dans le système d'information, le langage selon lequel ils seront décrits, la dynamique des opérations, et aussi le partage du travail entre l'automate et l'être humain.

Ces étapes de la démarche se rattachent toutes à un même point de vue sur l'entreprise, donc à une même définition qui, condensant ce point de vue, se rapporte à une *intention* et en définitive à des *valeurs*. Si cette définition ne peut prétendre fournir une représentation exhaustive de l'être qu'elle vise, il se peut que l'intention, les valeurs auxquelles elle se rapporte soient pourtant fécondes.

L'entreprise est le lieu où le travail des êtres humains s'organise afin d'agir sur la nature pour obtenir des résultats utiles.

Déployons cette définition pour l'analyser :

1) *Des résultats utiles* : le résultat de l'action, ce sont des *produits* (biens et services) utiles pour le consommateur ou pour la fonction de production d'autres entreprises. Ceci s'applique aussi bien aux administrations, qui produisent un service public⁶, qu'aux sociétés qui produisent des biens et services marchands.

Si le résultat de l'activité de l'entreprise n'est pas utile elle disparaîtra bientôt faute de clients, sauf si elle est maintenue sous perfusion par des pouvoirs qui la financent ou si, étant en fait une institution non pas productive mais prédatrice, elle prospère en détruisant de l'utilité.

Le rôle social de l'entreprise, c'est de *produire de l'utilité* : la compétitivité et le profit en sont des conséquences. Si le marché est organisé de façon à interdire la prédation, la recherche du profit sera le moteur de l'innovation (voir page 172).

2) *Agir sur la nature* : nous prenons dans cet ouvrage le mot « nature » au sens large qui englobe tout ce qui peut être obstacle ou moyen pour l'action et qui inclut donc, outre la nature physique, les natures sociale et humaine.

L'entreprise agit sur la nature en absorbant des matières premières qu'elle transforme en produits utiles⁷. La « physique de l'entreprise » recouvre toutes les tâches qui concourent à la production d'utilité : conception des produits, marketing⁸, achats, production physique au sens strict, commercialisation, distribution, service après-vente etc.

L'action sur la nature suppose un processus interne à l'entreprise. Ce processus est parfois tellement compliqué que les acteurs peuvent oublier sa finalité pour ne plus percevoir que sa procédure. Lorsque l'entreprise est bien rodée il arrive ainsi que l'attention accordée à sa physique s'estompe, au sein de la direction générale, pour faire place à la symbolique des territoires de légitimité, les « plates-bandes » des directions. Il peut en résulter des erreurs au plan physique. L'approche sociologique, lorsqu'elle s'arrête à ces symboles et néglige la physique de l'entreprise, est aveugle aux conditions pratiques de l'efficacité.

3) *Le travail des êtres humains s'organise* : l'entreprise met en œuvre le travail des êtres humains soit de façon différée (quand il est incorporé à un

6. Le service public fournit des *externalités positives*, produits utiles mais qui ne se prêtent pas à l'échange marchand et dont la distribution est donc « externe » au marché.

7. L'évaluation de l'utilité de la production doit donc tenir compte de la *désutilité* que comporte la destruction des ressources naturelles et des produits intermédiaires consommés lors de la production.

8. Nous prenons le mot *marketing* non pas dans son sens courant, qui est à peu près synonyme de *publicité*, mais dans son acception scientifique d'*analyse des besoins des clients*.

stock, le « capital fixe »), soit de façon immédiate (quand il contribue au flux du processus de production).

L'être humain qui travaille dans l'entreprise n'est pas un individu isolé. Sa compétence s'articule à celle d'autres personnes. C'est l'organisation de ce réseau de compétences qui procure à l'entreprise l'aptitude à l'action. Les compétences se forment, s'ajustent, se complètent selon un processus délicat.

La gestion des compétences suppose le *respect* envers l'être humain, son écoute attentive. Le modèle hiérarchique, quand il sacralise les fonctions de décision légitimes, néglige cette obligation et cela cause un gâchis de compétences.

L'entreprise parmi les institutions

Une *institution*, c'est une forme d'organisation qui, en vue d'une mission, obéit à des règles qui ont été *instituées* - et qui, de ce fait, se forment et évoluent dans l'histoire (Hatchuel [80]).

L'entreprise, forme d'organisation de la production des biens et services, est une institution au sens général du terme ; chaque entreprise, concrétisation particulière de cette forme d'organisation, est également une institution, mais au sens concret du terme.

Il existe des institutions qui ne sont pas des entreprises car leur mission n'est pas de produire des biens et services : la famille, l'Église, l'État, le droit, la médecine, la science etc.

Mais toute institution qui produit des biens et des services mérite d'être considérée comme une entreprise car l'élaboration d'un produit pose partout, *mutatis mutandis*, des problèmes d'organisation et de gestion analogues notamment en ce qui concerne l'informatisation, thème de cet ouvrage.

La frontière de l'entreprise ainsi définie ne coïncide pas avec celle de la propriété privée ni avec celle du marché : des administrations comme l'appareil judiciaire, l'armée, l'éducation nationale, les hôpitaux, voire même des structures comme le système éducatif, le système de santé etc., qui produisent des services non marchands, relèvent de la catégorie de l'entreprise telle que nous la considérons ici.

Ceux qui ne veulent voir dans l'entreprise que le lieu où la force de travail se fait exploiter estiment que lui accorder une influence sur la formation des adolescents serait préparer ceux-ci à la servilité. Si par contre on voit dans l'entreprise le lieu de l'action organisée, on estimera que préparer les jeunes gens à l'entreprise n'est rien d'autre que les préparer à l'action, et donc à l'âge adulte.

Dans les entreprises on affectionne souvent les métaphores maritimes : « nous sommes tous dans le même bateau », « il faut tenir le cap » etc. Nous allons recourir à une de ces métaphores pour faire apparaître l'écart entre le rôle économique de l'entreprise et son fonctionnement. Considérons une liai-

son maritime entre deux ports séparés par une grande distance. La fonction économique de cette liaison est de transporter des marchandises d'un port à l'autre dans des conditions admises de délai et de sécurité, mais pour que cette liaison puisse fonctionner il faudra des navires conçus selon l'art de la construction navale, et il faudra qu'un équipage gouverne chaque navire selon l'art de la navigation. Cela suppose des compétences, une organisation, un rapport avec la nature que l'expression « transporter des marchandises d'un port à l'autre » ne résume ni ne contient explicitement.

De même, les conceptions de l'entreprise qui la réduisent à la production du profit, à la « création de valeur » pour l'actionnaire, ignorent la façon dont l'entreprise s'organise et fonctionne ; l'aborder du point de vue de ses processus du production, comme on le fait lorsque l'on examine son système d'information, fait par contre entrer dans l'intimité de son organisation et de son fonctionnement.

4.2 Que produit-on ?

Le mot « production » a, en économie, un sens qui s'écarte de l'usage courant. Le vocabulaire des économistes présente cependant d'importants avantages.

Dans le langage courant, et jusque dans les normes validées par l'AFNOR et qu'appliquent les informaticiens, on distingue les « produits » et les « services ». Dans cette optique, un « produit », c'est ce qui se touche avec les mains, qui s'achète dans un magasin, que l'on emporte chez soi pour le consommer ou l'utiliser : une boîte de lessive, un logiciel, une automobile. Le logiciel s'achète sous la forme d'un CD-Rom accompagné d'une notice et emballé dans une boîte en carton. Il peut aussi être téléchargé *via* l'Internet, mais il reste un « produit » parce que *matériellement* il s'inscrit dans la mémoire de l'ordinateur.

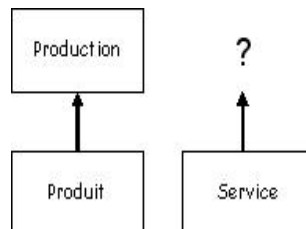


FIG. 4.1 – Vocabulaire courant : une image déroutante

Distinguer ainsi « produits » et « services », cela suggère que les « services » ne sont pas des « produits » et, par continuité sémantique, que n'étant pas *produits* ils ne résultent pas d'une *production* (figure 4.1). Alors que produire, ce serait élaborer des choses qui ont un volume, un poids, une consistance matérielle, les services relèveraient d'un *immatériel* qui semble flotter en l'air comme une vapeur : on ne sait pas au juste en quoi ils consistent.

Nous retiendrons la définition suivante des services :

Mettre à la disposition temporaire d'un client soit un bien (location d'un appartement ou d'une voiture), soit une capacité intellectuelle (conseil, expertise), soit un savoir-faire (dépannage, service après-vente), soit une combinaison de plusieurs de ces éléments (un billet d'avion donne le droit d'occuper une place à bord et de bénéficier du savoir-faire du pilote) (Demotes-Mainard [45]).

Les économistes, eux, utilisent le mot « produit » pour désigner l'ensemble formé par les « biens » et les « services » et ils appellent « bien » ce que le langage courant nomme « produit ».

Classer les services dans la catégorie des produits leur permet de *penser* la production des services. En effet pour la théorie économique, « produire » c'est essentiellement produire de l'*utilité*, de la *satisfaction* pour le consommateur. Alors le « produit » n'est pas seulement le bien matériel que l'on voit posé sur l'étagère du magasin. Ce qui est produit c'est, outre la fabrication du bien proprement dite, l'ensemble des services de conception, transport, commercialisation, distribution, avant et après-vente qui l'entourent et qui contribuent à son utilité : on ne peut pas en effet produire de la satisfaction sans mettre le bien entre les mains du client, et dans des conditions telles que celui-ci en soit satisfait (figure 4.2).

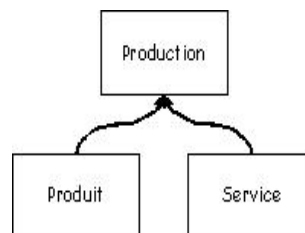


FIG. 4.2 – Vocabulaire des économistes : est produit tout ce qui contribue à l'utilité

Quand le vocabulaire sépare les concepts de « produit » et de « service », il est difficile de dire que l'on produit un service. Mais plus de 75 % de la population active française travaille dans le tertiaire : éducation, santé, commerce, transport etc. (Marchand et Thélot [201]). En suggérant que les services ne sont pas des produits, on s'engage sur la pente sémantique glissante qui aboutit à dire que 75 % de la population active ne produisent rien. On aura en outre du mal à penser la production de services, à l'organiser avec la même clarté d'esprit, la même énergie, la même bonne conscience que celle que l'on consacre à la production des biens. L'immatériel, même s'il occupe les trois quarts de la population, reste impensable. On parle de « qualité de service », mais on est bien en peine d'associer à cette notion une mesure et de se fixer des objectifs. Certaines entreprises se débattent

au fond de ce piège. Regardez cet opérateur télécoms : il s'est organisé par « produit » en créant des filiales distinctes pour la téléphonie fixe, la téléphonie mobile et l'Internet, et chacune envoie à chaque client plusieurs factures différentes ; il sous-traite à d'autres entreprises les travaux à faire chez les clients et rate ainsi l'occasion de contacts commerciaux fructueux. Quand on ne peut pas penser en termes de qualité de service, il ne reste plus, pour être compétitif, qu'à pratiquer la concurrence par les prix. Baissons les prix, la part de marché croîtra ! Oui, mais la marge unitaire diminue et l'on se rapproche de la stratégie fameuse « je vends à perte, mais je me rattrape sur la quantité ».

La concurrence par les prix s'appuie d'ailleurs sur une illusion d'optique, les mots se substituant aux choses. Si « un poulet égale un poulet », il faut bien sûr acheter le poulet le moins cher ; mais si les poulets se distinguent par leur saveur, si un poulet n'égale pas un poulet, alors un poulet plus cher peut apporter davantage de satisfaction s'il présente un meilleur rapport qualité/prix. *Même s'il s'agit exactement de la même machine*, un réfrigérateur n'égale pas un réfrigérateur selon la façon dont il est documenté, installé, entretenu, maintenu par le celui qui le vend.

Une comptabilité nationale trop sommaire, répartissant la valeur entre volume et prix sans tenir compte de la qualité, ne considère que le nombre de réfrigérateurs et de poulets vendus. Elle est incapable de prendre en compte la qualité du service incorporée au bien, donc d'évaluer correctement le volume effectivement produit. On oublie trop, dans le calcul des indices, que leur théorie s'appuie sur la fonction d'utilité... Les indices « hédoniques », qui tiennent compte de la qualité, sont rares parce qu'ils sont difficiles à établir. Il en résulte, dans la comparaison des PIB entre pays, des distorsions que l'on ne sait comment corriger.

L'utilité du système d'information réside souvent dans une meilleure qualité de service : il apporte plus de transparence, plus de réactivité, des coûts de transaction plus bas etc. Mais on ne peut la percevoir, et consentir l'effort nécessaire pour l'obtenir, que si l'on est averti de la contribution des services à l'efficacité de l'entreprise.

* *

Les entreprises qui satisfont le mieux le client sont celles qui s'organisent pour l'écouter et répondre à ses attentes. Elles ne sous-traitent à personne la relation avec lui ni le traitement de ses paiements par carte bancaire. Elles accordent beaucoup de soin à la qualité de la facturation comme à celle du plateau téléphonique, dont le personnel est compétent et expérimenté. Elles analysent les données collectées sur leur clientèle et la segmentent pour définir leur offre et leur démarche commerciale : la segmentation est découpée non selon leurs produits mais selon la nature des besoins des clients. Ces entreprises savent que leur production comporte non seulement l'élaboration physique du produit (largeur de bande, taille mémoire, rapidité du processeur, vitesse et confort de la voiture etc.) mais aussi l'aptitude du vendeur à comprendre ce que disent des personnes diverses et à leur répondre de façon compréhensible, la disponibilité du réparateur, la clarté des indications sur la nature et le délai des prestations, la ponctualité.

Tant que l'on sépare les concepts de « produit » et de « service », tant que l'on croit possible de vendre un bien sans l'accompagner du service nécessaire à la satisfaction du client, tant que l'on ne conçoit pas que la production du service doit être organisée avec le même soin que la production du bien, les centres de recherche se focaliseront sur la prouesse technique et négligeront la connaissance des besoins, la conception de la commercialisation, l'organisation du service de proximité, l'ingénierie d'affaires (montage de partenariats), alors qu'ils sont pourtant nécessaires pour *produire de la satisfaction*.

4.2.1 À propos de l'immatériel

« Treating intangibles as services obscures not only the real nature of intangibles but also that of services » (Peter Hill, « Tangibles, Intangibles and Services », *Conference on Service Centre Productivity*, Ottawa avril 1997.

On dit que l'économie actuelle est *immatérielle*. Ce terme évoque quelque chose de vaporeux, d'impalpable qui fait obstacle au raisonnement. Par ailleurs, on assimile parfois l'immatériel aux « services » ou encore à l'« innovation ». Mais l'essentiel d'un service réside dans le caractère temporaire de sa mise à disposition (cf. page 158), non dans son caractère immatériel ; et il entre par ailleurs une part d'immatériel dans un bien : toute machine a dû être conçue avant d'être fabriquée et sa production industrielle concrétise, en un nombre d'exemplaires éventuellement élevé, une *conception* qui a cheminé de l'idée floue initiale jusqu'au plan précis, puis aux méthodes et à l'organisation de la production. Ainsi l'immatériel « mord » sur le terrain des biens comme sur celui des services et sa frontière ne coïncide pas avec celle qui les sépare (figure 4.3).

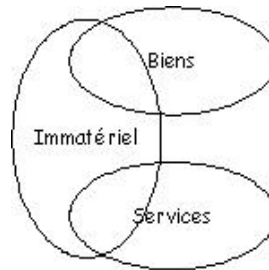


FIG. 4.3 – Biens, services et immatériel

Il ne convient pas non plus d'identifier l'immatériel et l'innovation : l'organisation d'une entreprise, ainsi que sa dimension juridique qui se concrétise par des statuts et des contrats, sont immatérielles sans pour autant résulter nécessairement d'une innovation : elles reproduisent en général des procédés éprouvés.

Ce qui est immatériel dans un bien comme dans un service, c'est le produit d'un travail de *conception*, des efforts qui *préparent* la production :

plans et procédés de fabrication ; organisation ; spécification et définition d'un programme informatique ; négociation et mise au point des contrats ; études de marketing ; définition du réseau de distribution. La conception est parfois une innovation, mais pas toujours : c'est pourquoi l'immatériel et l'innovation ne coïncident pas.

Prenons l'exemple d'un programme informatique. Son écriture proprement dite est précédée par la définition des spécifications fonctionnelles, puis par celle des contraintes techniques qu'il devra respecter pour pouvoir être intégré au système d'information. Cette phase préalable nécessite des consultations et négociations. Pendant l'écriture du code s'opère encore une mise au point. Le logiciel ne pourra être exécuté par l'automate qu'à l'issue de ces étapes de conception.

Le résultat des travaux de conception qu'une entreprise réalise (juridique, organisation, conception des produits et méthodes de production, connaissance des clients, gestion de la compétence) constitue un *stock* dont l'accumulation précède la production. L'immatériel *relève donc de la catégorie du capital*.

Pour certaines entreprises la production elle-même est immatérielle : c'est le cas de celles qui produisent des logiciels qu'elles vendent ou des brevets qu'elles commercialisent. Cela n'empêche pas ces produits immatériels de relever de la catégorie du capital, tout comme les machines-outils qui sont produites par certaines entreprises et destinées au capital d'autres entreprises.

* *

La mesure du capital immatériel est notoirement délicate mais pas sensiblement plus que celle des actifs matériels, dont on sous-estime sans doute la difficulté : les méthodes d'évaluation des actifs matériels sont diverses (selon que l'on adopte la démarche du comptable, du créancier ou de l'actionnaire, on prendra soit la valeur historique diminuée des amortissements, soit la valeur de remplacement, soit la valeur sur le marché de l'occasion etc.) et les fourchettes d'imprécision sont larges.

La difficulté de la mesure du capital immatériel vient plutôt des habitudes. Lorsque l'économie était dominée par la mécanique et la chimie, chaque effort de conception se concrétisait dans une machine ou dans une usine et on pouvait croire que l'on avait convenablement comptabilisé le capital immatériel en évaluant les machines, les bâtiments etc. Mais il n'en est plus de même aujourd'hui.

En effet l'économie automatisée a permis de valoriser les efforts de conception séparément de leur concrétisation dans un produit. Les programmes informatiques en sont un exemple : le droit d'utilisation d'un logiciel peut être commercialisé sur le réseau indépendamment de tout support matériel. Le « plus » économique apporté par l'ingénierie d'affaires en donne un autre exemple, ainsi que l'organisation des réseaux de distribution, la qualité de la gestion ou de la décision stratégique etc.

Cette nouvelle économie est une « économie de l'immatériel » en ce sens que c'est une économie de la conception (des circuits intégrés, des logiciels,

des ordinateurs etc.), la part du coût de conception dans la fonction de coût étant devenue majoritaire par rapport au coût de la production physique en quantité : c'est évidemment le cas pour les logiciels et les circuits intégrés, c'est aussi le cas pour des biens comme les automobiles et les avions qui comportent de plus en plus d'électronique et de programmes informatiques, et dont la conception s'appuie intensivement sur des simulations par ordinateur.

4.2.2 Pour une économie de la qualité

« Se former une idée claire des besoins sociaux et s'efforcer de la répandre, c'est introduire un grain de levain nouveau dans la mentalité commune ; c'est se donner une chance de la modifier un peu et, par suite, d'incliner, en quelque mesure, le cours des événements, qui sont réglés, en dernière analyse, par la psychologie des hommes » (Marc Bloch, *L'étrange défaite*, Gallimard 1990, p. 205)

La crise économique des années 1930 résultait, selon Keynes, d'une erreur collective d'anticipation (Dostaler [49]) : les entreprises, tout comme les consommateurs, sous-estimaient le potentiel productif nouveau qu'avait procuré la mécanisation de l'industrie. Le pessimisme des agents économiques bloquait l'économie dans une conjonction paradoxale de pénurie et de sous-emploi.

La crise larvée que connaissent aujourd'hui les anciens pays industriels ne résulte-t-elle pas d'un phénomène analogue ? Ne sous-estimons-nous pas le potentiel de l'économie *automatisée*, qui s'appuie sur les techniques de la microélectronique et du logiciel et sur l'informatisation des entreprises ?

Cette sous-estimation s'accompagnerait d'un défaut dans la perception de ce qui constitue, aujourd'hui, la richesse. Pas plus aujourd'hui qu'hier la richesse ne réside dans la production de profit, d'argent : elle réside dans la production d'utilité, de satisfaction pour le consommateur.

Mais on identifie souvent la richesse avec la production *en quantité*, la consommation *en quantité*, comme si nos pays riches connaissaient encore la pénurie. Produire plus d'automobiles, de chaussures, de meubles, construire davantage d'immeubles, ce serait être plus riche. C'est à cette conception de la richesse, héritée de la période de pénurie de l'immédiat après-guerre, que correspondent la mesure du Produit Intérieur Brut en volume et la mesure de la « croissance ».

Or à l'automatisation de la production a correspondu un changement de la fonction de coût : alors qu'auparavant le coût de production était fonction croissante de la quantité produite, il n'en dépend *pratiquement* plus aujourd'hui. Dès lors la valeur se détache de la quantité pour s'accoler à la *qualité*, à l'adéquation du produit aux besoins du consommateur, fût-ce dans un étroit segment de marché ; la diversification de l'offre, la différenciation des produits, leur adaptation qualitative aux divers segments importent davantage que le volume produit ou consommé.

Cette *économie de la qualité* caractérise depuis longtemps des biens comme les livres, les disques etc. : l'utilité que vous apporte un livre n'est pas accrue si l'on vous en donne un deuxième exemplaire, identique au premier, et n'est donc pas fonction de la quantité consommée. Mais ce qui caractérise l'économie automatisée, c'est l'extension de la place prise par la qualité y compris dans des domaines - nourriture, habillement, logement etc. - où la quantité semblait auparavant primordiale.

* *

Pour se représenter le rôle de la qualité dans l'économie, il est utile de parcourir l'histoire de la production que nous évoquerons ici dans ses très grandes lignes.

La standardisation fut pratiquée dès l'antiquité dans l'architecture, la production des armes, le textile, la construction navale (Adam [3]). Il en est résulté parfois, dans l'architecture romaine comme dans celle des églises gothiques, une médiocrité qui surprend ceux qui s'attendent à toujours trouver de la beauté dans l'ancien.

La production n'est devenue industrielle qu'au début du XIX^e siècle, les progrès de la métallurgie ayant alors permis de produire des machines efficaces - notamment la machine à vapeur, première en date parmi les moteurs qui supplanteront la force motrice humaine et animale ainsi que les moulins à vent ou à eau.

La mécanisation de l'industrie a procuré une baisse du coût de production qui lui permit de concurrencer victorieusement l'artisanat. Cependant la *conception* des produits de l'artisanat incorporait un très ancien savoir-faire qui leur avait conféré commodité, solidité et parfois beauté. L'industrie n'eut qu'à puiser dans le patrimoine ainsi accumulé pour définir ses premiers produits, les adaptant toutefois pour en faciliter la fabrication. Les produits du XIX^e siècle étaient moins chers, plus hygiéniques et plus commodes que ceux du XVIII^e - il suffit pour s'en convaincre de comparer un appartement bourgeois à un hôtel particulier aristocratique - mais la substitution de l'industrie à l'artisanat s'accompagna d'un affadissement du goût, la création artisanale n'étant plus là pour renouveler la conception. On a pu dire ainsi que le XIX^e siècle n'avait pas de style (si ce n'est celui de la commodité) parce qu'il les a tous copiés, du classicisme hellénique au baroque, dans un oppressant désordre esthétique (Mignot [134]).

Dans les années 1920 le « design » moderniste a réintroduit de la beauté dans la production industrielle. L'architecture, avec notamment le Bauhaus (1919-1933)⁹, le mobilier, l'équipement ménager, l'automobile, le vêtement furent alors repensés dans un souci de qualité et de fonctionnalité qui, tout en tirant parti de l'efficacité industrielle, renouait avec le meilleur de la démarche artisanale.

Si la conception s'est ainsi améliorée, l'économie industrielle est restée fondée sur la production massive de produits standardisés s'appuyant sur la mécanique, la chimie et la division du travail au sein d'une main d'œuvre

9. www.bauhaus.de/bauhaus1919/index.htm

nombreuse constituée d'individus interchangeables. La distribution elle aussi massive des produits passa par ces grands magasins dont Émile Zola a dès 1883 décrit l'essor dans *Au Bonheur des Dames*.

L'équilibre propre à l'économie mécanisée n'a cependant pu s'épanouir que dans les années 1950, après des crises et des guerres que l'on peut considérer comme autant d'épisodes d'adaptation. Par ailleurs la construction de cette économie s'est accompagnée, comme il se doit, de celle d'une structure institutionnelle qui lui était adéquate : organisations patronales et syndicales, droit et fiscalité, justice et police, sécurité sociale, école et formation professionnelle, santé et retraite, organisation des armées etc. Ces institutions, aujourd'hui encore, balisent notre vie collective et structurent notre imaginaire.

Cependant, et sans bien sûr que le système technique mécanisé ne disparaisse du jour au lendemain, celui-ci a perdu dans les années 1970 sa prééminence au profit du système technique *informatisé* (Gille [66]) : à condition qu'on le relie à des périphériques convenables (interface homme machine, bras d'un robot, avion en pilotage automatique etc.), l'automate permet de programmer tout ce qu'un automate peut faire. En outre le réseau a apporté l'ubiquité à cet automate programmable.

* *

Le passage d'un système technique à l'autre, progressif mais rapide, a suscité des changements que masque la continuité de la vie quotidienne. L'automatisation de la production physique a supprimé des postes de travail et dénoué les solidarités qui, jadis, s'étaient nouées entre l'emploi et la production industrielle, entre salaires et débouché de la production. Ce sont là des facteurs de crise auxquels les institutions, bien adaptées au système mécanisé qui leur avait donné naissance, s'avèrent incapables de répondre.

Par ailleurs la part de la conception et du dimensionnement dans le coût de production est allé en croissant : on pense ici d'abord aux réseaux télécoms, à la production de logiciels et de circuits intégrés, mais aussi à tous les produits dont la production a été automatisée. Le marché s'équilibre alors sous un régime de *concurrence monopoliste* (Volle [213]) qui déconcerte des raisonnements et des institutions construits autour de l'opposition polaire entre concurrence et monopole.

Sous le régime de la concurrence monopoliste, chaque produit subit une diversification qui l'adapte aux besoins de divers segments de clientèle. On sort ainsi du règne de la production de masse, quantitative, pour entrer dans celui de la *diversification qualitative*. La valeur de la production, c'est-à-dire son utilité, se mesure non plus selon le volume produit mais selon la pertinence et la finesse de la diversification. La personne la plus riche n'est pas celle qui peut consommer le plus (en quantité) mais celle qui, ayant accès à la plus grande diversité de produits, peut y trouver ceux qui (en qualité) répondent le mieux à ses besoins. Il en est de même *mutatis mutandis* pour la richesse des nations.

La diversification des produits suppose un marketing attentif aux besoins des consommateurs, une organisation réticulée de la production (un réseau

d'entreprises travaillant en partenariat pour produire un assemblage), une intermédiation qui aide le consommateur à trouver la variété qui lui convient le mieux, des services d'avant-vente et d'après-vente qui l'aident à tirer parti du produit : l'ensemble de cet édifice est fondé sur le système d'information.

Au changement de système technique correspond un changement de la mesure de la valeur économique, et aussi sans nul doute un changement des *valeurs* au sens philosophique et culturel du terme.

La quantité reste présente, fût-ce sous la forme de l'anticipation d'une demande aléatoire, dans le dimensionnement des réseaux (routes, télécoms, transport, énergie), et aussi dans la consommation : même si chacun n'achète qu'un exemplaire d'un même livre, il n'est pas indifférent pour l'éditeur de savoir combien d'exemplaires en sont vendus. Le prix étant le plus souvent attaché à l'unité, c'est enfin sur la quantité vendue que s'établit l'équilibre économique d'un produit¹⁰.

Le raisonnement économique auquel invite l'automatisation s'appuie ainsi sur un édifice conceptuel délicat. L'économie à coûts fixes, et la concurrence monopoliste qui en est le corollaire, en forment le premier étage. Puis vient l'économie du dimensionnement propre aux réseaux qui, elle, n'est qu'à moitié à coût fixe - puisque le coût est fixe à court terme une fois le réseau construit, mais varie à moyen terme en fonction de la demande anticipée. Enfin vient l'économie des services qui relève pour une part du coût fixe (car elle demande un travail de conception), pour une part du dimensionnement (on met en place le réseau de distribution du produit en même temps que l'on organise sa fabrication), pour une part du coût variable (les services d'avant et après-vente demandent d'autant plus de travail que l'on vend davantage d'unités du produit).

* *

L'emploi que l'automatisation a chassé de la production physique se réfugie dans les services. Dans les pays riches, plus des trois quarts de la population active travaillent aujourd'hui dans le tertiaire : c'est un fait que notre imaginaire peine à assimiler, tant la notion de production reste accolée à la production physique. Parmi les blocages dont souffre notre économie et qui l'empêchent d'atteindre sa pleine efficacité, la réticence à développer les services est l'un des plus tenaces.

Cette réticence provoque une perte d'utilité sensible pour le consommateur. La qualité d'une photocopieuse, par exemple, dépend moins de la machine (toutes les marques produisent des machines équivalentes depuis que les brevets de Xerox sont tombés dans le domaine public), mais de la rapidité du dépannage en cas d'incident. Lorsque vous faites le plein dans une station d'essence en libre-service, cela vous prive de l'aide de quelqu'un qui examinerait la pression et l'usure de vos pneus, vérifierait les niveaux, nettoierait le pare-brise et contribuerait ainsi à votre sécurité.

10. D'autres formes de tarification sont cependant possibles, par exemple un droit d'accès forfaitaire.

Si l'on a supprimé ces emplois-là, c'est en raison d'une conception de la « productivité » qui se détourne de la satisfaction des besoins du client, de la production d'utilité. Il est fréquent, et très désagréable, d'attendre longtemps un dépannage, de se faire « remettre à sa place » par un opérateur de centre d'appel, par un employé retranché derrière son guichet, par un chef d'atelier ou un conseiller d'agence bancaire mal lunés. Il ne faut pas s'en prendre à ces personnes mais à l'entreprise qui les a mal outillées, mal formées et mal encadrées. Le client maltraité regimbe rarement mais quittera cet opérateur télécoms, ce fournisseur de télécopieurs, cette marque automobile, cette banque, s'il trouve ailleurs un service plus efficace.

* *

La qualité de service a un coût : l'entreprise qui offre un service de qualité ne peut donc pas être, en même temps, celle qui affiche le prix le plus bas. C'est pourquoi il y a quelque chose de pervers dans la multiplication des soldes et promotions, dans le fait que les chaînes de distribution utilisent, pour se faire concurrence, le seul argument du prix : « Mammouth écrase les prix », « Le n° 1 du prix chez Carrefour », « Auchan veille à toujours proposer les prix les moins chers » etc.

« Nous avons tout tenté, disent les responsables du marketing des grands magasins, mais le seul argument qui marche auprès des clients, c'est le prix ». Est-ce vrai ? Le consommateur peut, durant les périodes de transition qui sont aussi des périodes de désarroi, ne pas concevoir exactement ses propres besoins et obéir à des pulsions qui bloquent l'économie - comme, par exemple, le désir de se sentir plus « malin » qu'un autre. Pour profiter des indemnités que les transporteurs aériens offrent aux victimes du *surbooking*, certaines personnes s'arrangent ainsi pour arriver au dernier moment à l'enregistrement (Bouzou [24]). Cette médiocre « économie » leur fait perdre le temps d'un aller-retour vers l'aéroport : ces personnes-là ont-elles une notion exacte de leur propre utilité ?

Certains disent, avec l'apparence du bon sens et de la générosité, que la qualité ne peut avoir d'importance que pour les personnes à l'aise alors que les pauvres, eux, n'auraient besoin de rien d'autre que d'un prix bas. Ils croient être ainsi « sociaux » et peut-être même « de gauche ». Mais la qualité, notamment la qualité de service, n'est pas seulement le plus qui contribue au confort du bourgeois à l'aise. Elle réside dans l'identification du produit adéquat au besoin du client et dans les services (entretien, dépannage) qui le rendront utilisable. Que l'on ne prétende pas que les pauvres n'ont pas besoin de cette qualité-là !

On peut d'ailleurs distinguer deux sortes de qualités : la qualité « horizontale », qui résulte de la diversification de produits en variétés différentes ayant le même coût de production et vendues au même prix (chemises bleues et chemises roses), et la qualité « verticale » qui distingue divers degrés de finition, donc de coût et de prix (chemises de confection et chemises sur mesure). Certes, seuls ceux qui en ont les moyens peuvent s'offrir la qualité « verticale », mais pourquoi la refuser ?

Jamais l'automobile n'aurait pu naître s'il n'y avait pas eu des riches pour en acheter et aujourd'hui l'automobile est un produit de masse : la qualité « verticale » d'aujourd'hui préfigure la norme future, plus élevée, du produit courant. Faut-il donc que les modes vestimentaire et alimentaire des personnes aisées imitent, de façon symbolique, le style de vie des plus démunis ? N'est-il pas normal et sain que celui qui en a les moyens s'habille de façon soignée et se procure, lorsqu'il veut manger un poulet, non pas un poulet de batterie mais un savoureux poulet fermier dont il paiera le prix ? N'est-il pas normal et sain que les entreprises qui fabriquent des produits de qualité, et offrent les services correspondants, puissent rencontrer la demande de consommateurs avertis ?

Quand la publicité, assourdissante, se cale sur la seule longueur d'onde du prix le plus bas, quand l'entreprise néglige les services qui devraient accompagner son produit et ne se soucie pas de son système d'information, elles contribuent à répandre un modèle de consommation qui ne répond ni aux besoins des consommateurs, ni aux possibilités de l'appareil productif, et qui maintient l'économie dans une conjonction de sous-utilisation des ressources et de sous-emploi analogue, *mutatis mutandis*, à celle de la crise des années 1930.

4.3 Mission et organisation

« Quand la relation entre l'État et ses sociétés devient inconfortable, c'est souvent parce que l'État n'a plus de point de vue réel sur leurs missions, ne sait plus en inscrire les activités dans les perspectives de sa propre politique, n'arrive pas à arbitrer entre ce qu'il juge essentiel et ce qui lui paraît accessoire, n'accepte plus de faire des choix et se contente de saupoudrer par routine. En un mot quand la vision et la conviction font défaut. »
(Jean-Jacques Aillagon « J'attendais un minimum de soutien de l'État », *Le Monde*, 4 mars 2006.)

Quoi de plus *réel* qu'une maison ? Elle pèse de tout le poids des matériaux de construction. Elle occupe un volume qu'elle découpe en étages, couloirs et pièces. La circulation de l'air et des personnes est contrainte à passer par ses ouvertures. Mais en la considérant comme un objet physique, aurons-nous tout dit ? Non, car avant d'être construite la maison a été un *projet*. Le maître d'ouvrage et le maître d'œuvre ont dû discuter son plan. Certaines décisions ont été prises en cours de chantier. Résultant d'un projet, la maison pourra par la suite faire l'objet d'autres projets : déplacer une cloison, modifier le contour d'une ouverture, ajouter un étage, adjoindre un bâtiment latéral. À sa réalité physique massive est donc associée, sur un autre plan, la réalité des *intentions* qui ont suscité sa construction et susciteront ensuite des modifications. Ces intentions, notons-le, s'expriment non seulement dans l'organisation de l'espace mais aussi au plan symbolique : par ses proportions comme par ses détails la maison énonce, dans la langue de l'esthétique, un rapport au monde, une position sociale, une filiation culturelle.

La maison peut servir de métaphore aux artefacts que fabrique l'être humain et en particulier aux *institutions*, qu'il s'agisse de l'Église, de l'entreprise, du droit etc.

Toute institution répond à une intention, et remplit donc une *mission*. Pour pouvoir le faire effectivement, elle doit se doter d'une *organisation*. Nous appellerons *incarnation* la relation entre mission et organisation (figure 4.4) : il s'agit en effet de l'acte par lequel la mission prend chair et se dote du point d'appui physique nécessaire à sa réalisation¹¹. Une mission sans organisation reste une velléité : il faut une organisation pour concrétiser la mission, la mettre en œuvre, l'incarner dans le monde.

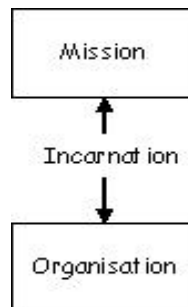


FIG. 4.4 – *L'incarnation de la mission*

Ceux qui sont familiers de la culture chinoise reconnaissent ici le couple que forment le Yáng (masculin, sud, soleil) et le Yin (féminin, nord, ombre), ou encore le Ciel Tian et la Terre Di : d'un côté l'impulsion, l'intention ; de l'autre la réalisation, la concrétisation (Jullien [94]). On retrouve ce même couple dans le judaïsme avec la relation entre le créateur et sa création. L'islam comme le christianisme en ont hérité, et l'incarnation de Dieu dans un être humain est le fait central du christianisme. Il s'agit donc d'une structure fondamentale, partagée par diverses cultures.

* *

Parfois la mission reste implicite. Mais quelle est la mission d'un transporteur aérien : faire voler des avions, transporter des passagers et du fret d'un aéroport à l'autre, fournir au client l'ensemble des services associés au transport aérien ? Quelle est la mission de l'ANPE : contribuer à l'intermédiation du marché du travail, apporter aux chômeurs une assistance psychologique, administrer le service public de l'emploi ? Quelle est la mission d'un opérateur télécoms : assurer la communication entre des équipements terminaux, fournir les services qui assurent et exploitent l'ubiquité logique ? La formulation de la mission est dans l'institution l'occasion de conflits confus, mais d'autant plus violents, entre des intentions antagoniques.

¹¹. On pourrait, certes, utiliser pour décrire ce processus d'autres mots que mission, incarnation et organisation : que le lecteur se sente libre de les remplacer par les termes qui lui conviennent le mieux.

L'organisation rassemble les moyens humains, matériels, financiers, et les procédures de travail nécessaires à la réalisation de la mission. Elle définit les biens et services finals qu'il s'agit de fournir, le réseau de distribution qui les mettra à la disposition des utilisateurs, les services support et biens intermédiaires qu'elle produira pour son propre usage. Elle structure les sphères de décision légitime en directions et services. Elle recrute et forme son personnel, définit ses processus de travail, édicte des normes, élabore un système d'information. Elle finance son capital fixe et son besoin de fonds de roulement. Elle s'installe dans des immeubles. Elle s'insère ainsi dans le monde de la nature (nature physique, mais aussi sociale et humaine) où réside la *physique de l'entreprise* (figure 4.5).

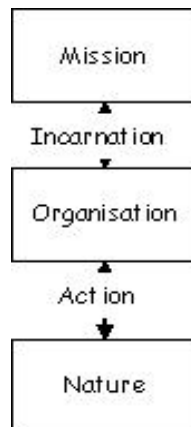


FIG. 4.5 – *L'action sur la nature*

Cela donne à l'entreprise une consistance solide, parfois impressionnante, qui remplit tout comme le fait la maison une fonction à la fois pratique et symbolique. Dans l'agencement, dans le décor et jusque dans l'odeur de son hall d'entrée, de ses couloirs, bureaux et salles de réunion, dans la tenue vestimentaire et la tonalité des conversations, enfin et surtout (si on peut l'examiner) dans son système d'information, l'institution transcrit sa mission selon une *personnalité* qui, pour ne pas être celle d'un être humain, n'en est pas moins particulière et reconnaissable tout comme peut l'être celle d'une ville.

L'incarnation se prolonge dans la durée à travers l'action de personnes qui font leur travail sans perdre de vue la mission et en s'affranchissant, quand c'est nécessaire, du caractère machinal des procédures : elles les interprètent en les respectant, tout comme un bon musicien interprète sa partition.

En écrivant ces lignes me reviennent à l'esprit les caractères admirables d'une secrétaire dans un ministère, d'un directeur à Air France, d'un autre à France Telecom, d'un conseiller d'éducation, d'un camarade de l'INSEE, d'une vendeuse de grand magasin, d'un chargé d'études dans une banque, de bien d'autres ! De telles personnes sont le sel de l'institution qu'elles

animent au sens exact du terme, car elles lui donnent une *âme*. Leur sérieux professionnel s'associe à une lucidité sans amertume et teintée d'humour. Ce sont elles, et non la hiérarchie, qui en constituent l'élite.

Parmi les dirigeants, ceux qui méritent le titre de *stratège* conjuguent ce même réalisme pratique à la même conscience de la mission ; ils y ajoutent l'art de former l'équipe de personnes de confiance qui leur permet de démultiplier leur action.

La fonction d'animation n'a pas de place dans l'organigramme, on ne sait le plus souvent aucun gré à ceux qui la remplissent, mais si elle est supprimée l'institution devient une machine aveugle dont la seule finalité réside dans la routine des procédures : l'organisation tend naturellement à s'émanciper de la mission.

Si en effet l'organisation est faite pour agir sur le monde dans lequel elle incarne la mission, la réalisation de celle-ci rencontre la complexité du monde. Inévitablement, cela nécessite des compromis : il faudra parfois agir sans avoir toutes les connaissances nécessaires, sans pouvoir anticiper les conséquences de l'action ; le mieux étant l'ennemi du bien, il faudra aussi tolérer des imperfections.

La confusion des valeurs qui en résulte éveille des tentations et fraie la voie à la *trahison* : l'organisation peut finir par agir d'une façon contraire à une mission qu'elle a oubliée. Il arrive ainsi que des dirigeants trahissent leur entreprise, que des salariés ou des syndicats agissent au rebours de l'efficacité, qu'une armée utilise ses armes pour opprimer son pays, que la justice soit plus formelle que soucieuse d'équité, que des médecins fassent passer leur corporation avant la santé des patients, que des entreprises soient indifférentes à leurs clients etc. Le risque de ces trahisons est le prix dont il faut accepter de payer l'incarnation : la mission ne pourrait rester parfaitement pure que si elle n'était aucunement mise en œuvre, mais alors ce serait comme si elle était annulée.

La tentation de la trahison est d'autant plus forte pour une personne que celle-ci dispose de plus d'initiative et d'autorité dans l'organisation. Dans une entreprise, seront fidèles à la mission la plupart des agents opérationnels, à peu près la moitié des personnes de la direction générale (l'autre moitié se consacrant aux délices de l'intrigue), mais seule une minorité parmi les dirigeants. Dans l'Église la trahison sera plus tentante, donc plus fréquente, parmi les prélats que parmi les religieux, parmi ces derniers que parmi les simples fidèles. Cela provoque des scandales douloureux. Certains croient en effet que si la mission était juste il devrait nécessairement en résulter une organisation impeccable : le fait que l'organisation trahisse leur semble invalider la mission elle-même. « Le gros de l'effort de recherche est orienté vers la conception des armes, le savoir-faire peut servir le mal, l'entreprise n'est pas rationnelle, l'administration est inefficace, le pape se trompe » : et les dénonciateurs en déduisent un peu vite que la science, la technique, l'économie, l'État, l'Évangile etc. ne valent rien ou sont même nuisibles.

D'autres personnes, que les dénonciateurs taxeront de naïveté ou d'hypocrisie, préfèrent pour s'épargner les douleurs du scandale ignorer jusqu'au risque de trahison. Elles veulent croire l'agent économique rationnel, le pape

infaillible, les dirigeants politiques capables, les magistrats équitables, les médecins dévoués etc. Elles énoncent des phrases comme « je suis fier de porter l'uniforme de notre armée » ou « je fais confiance à la justice de mon pays ».

Quand on a compris que la mission s'incarnait dans l'organisation, et constaté les compromis et tentations dont se paie l'incarnation, quand on sait que dès qu'il y a tentation la probabilité de la trahison n'est pas nulle, on conçoit que pour évaluer une organisation il faudra dépasser son aspect mécanique pour revenir à sa mission sans se laisser impressionner par les prestiges de la hiérarchie des pouvoirs légitimes.

Ici cependant se rencontrent deux difficultés. La première réside dans la définition de la mission elle-même qui, nous l'avons dit, est sujette à discussion. Nous avons considéré l'entreprise comme « le lieu où le travail des êtres humains s'organise afin d'agir sur la nature pour en obtenir des résultats utiles » : cette définition-là implique celle de la mission. Mais d'autres diront que la mission de l'entreprise, c'est de croître, de faire croître le cours de ses actions, de « produire de l'argent », de distribuer des dividendes, de « faire de la finance » : nous leur répondrons page 189.

D'autres encore diront que la mission de l'entreprise, c'est de « créer des emplois ». On ne peut, certes, lorsque le chômage est élevé, éprouver aucune sympathie pour les entreprises qui semblent vouloir licencier le plus grand nombre de salariés possible, ni pour les actionnaires qui leur font fête. L'entreprise doit respecter les personnes qu'elle emploie (voir page 177), développer et organiser leurs compétences. Mais si la mission de l'entreprise est de produire efficacement des choses utiles, elle ne peut pas être en même temps de créer des emplois : il faut plutôt dire qu'il revient au système éducatif (constellation qui rassemble l'institution scolaire, la formation professionnelle, la famille et les médias) de former des personnes employables.

Arrêtons-nous un instant sur le paragraphe qui précède car sa sécheresse peut contrarier des sentiments politiquement corrects. On ne doit pas exiger de l'entreprise - pas plus que de l'école - qu'elle soit une garderie pour des personnes qui n'ont rien à y faire. La meilleure contribution que l'entreprise puisse apporter à l'équilibre du marché du travail, c'est son efficacité et la qualité de ses produits. Il revient aux autres acteurs - et, en tout premier, au système éducatif - d'apporter leur propre contribution.

La deuxième difficulté est plus fondamentale. Beaucoup de personnes croient que pour être rigoureux, pour être scientifique, il faut se borner à décrire l'organisation telle qu'elle est : elles appellent cela *objectivité*. Se référer à la mission, qu'il est d'ailleurs si difficile de définir, ce serait être *normatif* et pécher contre l'*objectivité*.

Certains pensent en effet que dans les choses humaines, tout comme dans le monde de la nature, ne jouent que des mécanismes aveugles¹². L'intelligence que l'on croit discerner dans l'agencement des institutions résulterait d'une sélection naturelle. Le courage, la loyauté, la lucidité, le discernement seraient illusoire et sans conséquences. Se référer à une intention, à un pro-

12. C'est la thèse de Tolstoï dans *La Guerre et la Paix* (1872).

jet, serait aussi fallacieux que l'*intelligent design* que ceux qui prennent la Bible à la lettre tentent d'opposer à la théorie de Darwin.

Cette représentation mécaniste et fataliste plaque abusivement, sur le délai de l'action humaine qui se mesure en jours ou en années, une théorie qui, concernant l'évolution biologique, considère le délai nécessaire pour qu'une mutation génétique ait un effet significatif, soit une dizaine ou centaine de générations.

Il m'est arrivé d'évoquer devant un sociologue connu les défauts de l'organisation d'une entreprise pour laquelle je travaillais et qu'il avait lui-même étudiée. « Qu'est-ce qui vous autorise, s'écria-t-il avec colère, à dire que cette entreprise fonctionne mal ? Elle est comme elle est. Personne ne peut définir ce que c'est qu'une entreprise qui marche bien : une telle norme relèverait du Café du Commerce ». Pourtant, lui ai-je répondu, si l'entreprise est un être vivant, ne peut-il pas arriver qu'elle soit malade ? Et pour diagnostiquer la maladie et prescrire un traitement, ne faut-il pas avoir quelque idée de ce qu'est la santé ?

S'il s'agit d'ailleurs d'observer et de décrire, pourquoi se limiter à l'organisation ? À l'échelle de temps de l'action humaine la mission est elle aussi réelle et observable, même si elle n'est pas physiquement manifeste, même si elle n'est pas explicite. Elle est présente, fût-ce obscurément et en toute ambiguïté, dans les têtes des personnes que l'institution emploie comme dans les attentes de celles qui lui sont extérieures. Ces représentations orientent leurs intentions. Que la mission soit (ou semble) trahie, et il en résultera confusion, désarroi, désorientation ; l'édifice des valeurs sera compromis, le chaos s'installera dans les esprits. Une description de l'institution qui se limite à son organisation, qui évite toute référence à la mission, reste donc partielle : et dès que la mission est évoquée, cela fournit une norme pour évaluer l'organisation, que l'examen du système d'information dévoilera tout comme la radiographie révèle les organes internes d'un être vivant.

4.4 Le moteur de l'innovation

La « conduite du changement » n'est rien d'autre que l'adaptation de l'entreprise à une innovation qui transforme ses produits, sa fonction de production, son mode de commercialisation et enfin son positionnement. La résistance au changement réside, croit-on, dans les habitudes des salariés : il faudrait les convertir à la nouveauté. L'expérience indique, bien au contraire, que l'endroit où la résistance au changement est la plus forte se trouve tout au sommet de l'entreprise, dans le comité de direction. Sans doute il n'est pas facile pour les agents opérationnels d'assimiler les nouveautés que le système d'information leur apporte mais ce n'est pas eux qui pendant des années ont refusé la messagerie, le Web, l'urbanisation du système d'information, l'Intranet, l'informatisation de la relation avec les clients, la construction de tableaux de bord lisibles etc.

Le cerveau individuel est le lieu de naissance de toute idée vraiment nouvelle : seul cet organe est capable d'assurer la synthèse d'un édifice théorique, depuis le choix des concepts jusqu'à l'énoncé des hypothèses qui les relie.

Il arrive que la même idée naisse simultanément dans deux cerveaux différents parce qu'elle était dans « l'air du temps » : ce fut le cas de l'invention simultanée du calcul différentiel et intégral par Newton et Leibniz, du téléphone par Graham Bell et Elisha Gray, de la commutation de paquets par Paul Baran et Donald Davies (voir page 332), du chiffrement à clé publique par James Ellis et Whitfield Diffie (voir page 369). Mais ils ne s'étaient pas concertés et même si leurs inventions furent simultanées elles n'en étaient pas moins individuelles.

Que devient l'idée nouvelle lorsque son inventeur l'exprime ? Parfois elle est immédiatement adoptée comme une évidence et alors elle sera vite banale. Le plus souvent, et surtout si elle implique des enjeux importants, elle est considérée comme une incongruité. Un silence réprobateur l'accueille¹³.

Toute institution est en effet soucieuse avant tout de stabilité. L'organisation est fragile, les savoirs qu'elle articule ont été définis de longue main, les programmes de formation et les processus de travail sont rodés, le système d'information est en place. L'idée nouvelle perturberait cet agencement délicat : elle est donc naturellement rejetée, parfois avec violence. Malheur à l'innovateur.

Les historiens qui ont décrit les origines de la machine à vapeur, du moteur à explosion, du moteur électrique, de l'informatique, de la commutation de paquets (voir page 336) ont tous évoqué ce phénomène : l'idée qui se condense dans le cerveau d'un individu est, le plus souvent, d'abord refusée. L'armée française a d'abord refusé l'avion, AT&T le réseau de données (voir page 332), IBM la communication entre ordinateurs (voir page 333), France Telecom le téléphone mobile et l'Internet, Xerox n'a su que faire des inventions du Palo Alto Research Center (voir page 113) etc.

La confrontation inégale entre le cerveau individuel et l'institution a quelque chose de navrant si l'on pense aux talents qui sont ainsi broyés - le talent de l'inventeur bien sûr, mais aussi celui des personnes, bien plus nombreuses, qui auront préféré bloquer leur cerveau pour ne pas prendre le risque de s'exposer à l'adversité.

Pourtant l'idée, aussi ingénieuse qu'elle soit, *ne portera ses fruits que si elle est mise en œuvre par une institution*. Le moteur à réaction, le TGV, le circuit intégré, les systèmes de réservation, la carte orange de la RATP : ces idées ne pouvaient avoir de conséquences pratiques que si des institutions industrielles, commerciales, financières se mettaient en branle pour les réaliser, les incarner dans le monde de la nature.

Il est dans l'ordre des choses que l'entreprise ne comprenne pas l'inventeur, n'admette pas l'invention. Ce qui est plus difficile à expliquer, c'est

13. J'entends encore le silence qui me répondit lorsque, en 1997, j'ai montré à une grande entreprise qu'elle était la seule de son secteur qui n'ait pas donné à ses clients la possibilité de passer commande *via* la Toile.

comment l'incompréhension finit par céder et pourquoi, malgré tout ce qui incite l'entreprise à rester identique à elle-même, il se produit des innovations et même finalement *beaucoup* d'innovations : AT&T a fini par s'intéresser aux réseaux de données, IBM à la communication entre ordinateurs, France Telecom au téléphone mobile et à l'Internet etc.

Comment bascule l'opinion d'un dirigeant ? Comment se condense la décision d'un comité de direction ? Quand et comment une proposition jusqu'alors inaudible devient-elle une évidence ? Il faudrait, pour le décrire en détail, plonger dans la sociologie de la prise de décision et aussi dans ce Saint des Saints de la conscience où se délimite l'évidence.

Nous ne considérerons ici qu'une seule des dimensions de l'innovation, la dimension économique : certes l'entreprise ne se réduit pas à la seule économie mais c'est sur le terrain de l'économie que se tranche, en dernier ressort, l'évaluation de son efficacité ; et l'efficacité, c'est l'enjeu de l'entreprise, le ressort des phénomènes sociologiques dont elle est le théâtre.

* *

Il semble facile d'expliquer l'innovation par l'économie : « si l'entreprise innove, c'est parce que l'innovation lui procure un profit ». Il y a du vrai dans cette proposition banale mais elle semble contredire les enseignements de la théorie économique - de cette partie du moins de la théorie qui est consacrée à la concurrence parfaite et qui pèse si lourd dans les premières années de sciences économiques, dans la spécification des modèles comme dans l'opinion des économistes eux-mêmes, car ils ne prêtent pas tous beaucoup d'attention aux théories de la concurrence imparfaite (Tirole [203]).

Selon la théorie de la concurrence parfaite, un équilibre à *profit nul* s'instaure lorsqu'un marché est servi par de nombreuses entreprises et que de nouvelles entreprises peuvent y entrer librement : s'il était possible d'y faire du profit, de nouvelles entreprises entreraient, l'offre augmenterait, le prix baisserait jusqu'à ce que le profit s'annule. Le profit étant nul, le prix est égal au coût moyen de production. Or on démontre qu'à l'équilibre le coût moyen de production est minimal. La concurrence parfaite apparaît donc comme la clé d'une efficacité tout entière au service du consommateur, puisque le produit lui est proposé pour le prix le plus bas possible¹⁴.

Il n'en est pas de même si le marché est non sous le régime de la concurrence parfaite mais sous celui du monopole. Il n'existe alors qu'une seule entreprise sur ce marché. Elle peut fixer le prix au niveau qui maximise son profit, plutôt qu'au niveau le plus favorable au consommateur.

C'est ce raisonnement, très simple, qui fonde les politiques en faveur de la concurrence. « Le monopole, c'est mal ; la concurrence, c'est bien ». De Rockefeller à Microsoft, des magnats ont fait de gros profits sur le dos des consommateurs : les législations anti-monopolistes ne manquent donc pas de

14. En fait toute entreprise est la réalisation d'un projet qui a fait l'objet d'anticipations incertaines ; le profit doit donc à l'équilibre rémunérer le risque pris par l'investisseur. Au lieu de « profit nul » il faudrait dire « profit normal », le profit normal étant égal au taux d'intérêt sur le marché monétaire augmenté de la « prime de risque » minimale jugée nécessaire en raison des incertitudes propres au projet considéré.

fondement. Toutefois si le marché est sous le régime du « monopole naturel » - c'est-à-dire si la fonction de coût du produit est à rendement croissant - il serait inefficace de s'opposer au monopole puisqu'il permet de minimiser le coût de production ; mais il faut contraindre l'entreprise, par la voie réglementaire, à partager les fruits de l'efficacité avec les consommateurs en pratiquant un prix raisonnable.

Les plus libéraux des économistes reconnaissent la nécessité d'une réglementation mise en œuvre par un « régulateur ». Cette agence du gouvernement a pour mission (a) de contraindre les monopoles naturels à pratiquer un prix raisonnable, (b) de promouvoir la concurrence dans les secteurs où n'existe pas de monopole naturel et d'éviter qu'un monopole de fait ne vienne s'y instaurer par des moyens frauduleux ou violents.

Mais ce raisonnement suppose que la définition des produits, ainsi que celles de la fonction de production et de la fonction de coût, soient pérennes. Or l'innovation introduit des produits nouveaux ou modifie la fonction de production des produits existants.

Lorsqu'une entreprise innove, c'est pour rompre le cercle dans lequel l'enferme la concurrence. Si elle crée un produit nouveau, c'est pour bénéficier d'un *monopole temporaire* et réaliser un surprofit pendant le délai qui sera nécessaire aux concurrents pour réagir par une offre comparable. Si par une innovation de procédé elle réduit son coût de production, c'est encore pour réaliser un surprofit pendant le délai qui sera nécessaire aux concurrents pour adopter le nouveau procédé.

Seule la perspective d'un surprofit explique que l'entreprise fasse l'effort pénible qui accompagne l'innovation : elle devra en effet payer le coût de l'investissement, réorganiser ses processus de production et circuits de distribution, acquérir des compétences nouvelles, redéfinir les missions et les contours des directions, tout cela pour un résultat incertain.

Le montant du surprofit dépend du délai pendant lequel l'entreprise pourra bénéficier d'un monopole. Si ce délai est trop court, l'effort de l'entreprise n'aura pas été payé de retour. Si ce délai est long, l'innovation aura suscité un surprofit durable.

Mais supposons que le délai soit suffisant pour que le surprofit rémunère exactement l'effort de l'entreprise. À la fin de ce délai, les concurrents auront riposté soit par une autre innovation, soit par l'imitation. Le profit sera revenu à son niveau normal. Le rapport qualité/prix du produit s'étant accru, le pouvoir d'achat du consommateur se sera accru d'autant. *Le consommateur sera, en fin de compte, le bénéficiaire de l'innovation.*

Si le surprofit a convenablement rémunéré l'effort de l'entreprise, celle-ci sera incitée à innover de nouveau. Alors s'amorce le moteur de l'entreprise innovante, que l'on peut représenter par un schéma à quatre temps (figure 4.6).

Le caractère *temporaire* du monopole est essentiel au fonctionnement de ce moteur. Pour que le surprofit récompense l'innovation, il faut que l'entreprise bénéficie d'un monopole ; mais *pour qu'elle soit incitée à innover de nouveau*, il faut que le monopole ne s'éternise pas. Si en effet l'entreprise bénéficiait d'un surprofit durable, pourquoi se donnerait-elle la peine d'in-

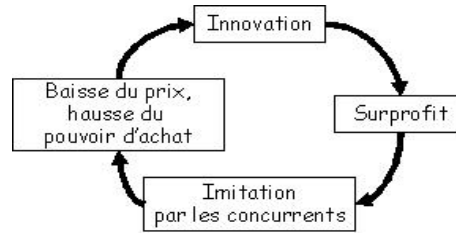


FIG. 4.6 – Moteur à quatre temps de l'entreprise innovante

noyer de nouveau? Un monopole durable ferait d'elle une rentière incapable d'évaluer les risques, donc de prendre le moindre risque.

* *

Considérons une économie qui, par les effets d'une réglementation bien conçue, d'un système de brevets bien pensé ou de tout autre mécanisme favorable, garantit aux entreprises innovantes le délai raisonnable, mais pas plus, pendant lequel elles pourront faire un surprofit. Alors s'amorce un type de croissance spécifique fondé sur l'amélioration renouvelée des produits ou la baisse renouvelée des coûts de production. Le bénéfice de l'innovation est, après chaque cycle, transféré aux consommateurs sous la forme d'une hausse de leur pouvoir d'achat.

Les modèles économiques de croissance les plus classiques (Intriligator [89]) postulent que la fonction de production et la nature des produits sont constantes. Si la fonction de coût est à rendement décroissant - hypothèse qui, au niveau macroéconomique, est vérifiée en raison de la hausse du prix des facteurs de production lorsque leur demande s'accroît - la croissance de la production s'aligne à terme sur celle de la population active et la production par tête se stabilise. Pour échapper à cette conclusion que l'expérience des XIX^e et XX^e siècles contredit trop visiblement, Solow [192] a introduit le progrès technique comme un résidu et Romer [175] a développé la théorie de la croissance endogène.

Le *moteur de l'entreprise innovante* correspond à une expérience familière aux entrepreneurs, chercheurs et ingénieurs, même si ceux-ci ne la formalisent pas. Il est vrai qu'il peut surprendre ceux qui sont habitués à raisonner sur une économie à l'équilibre: lorsque ce moteur tourne, l'économie n'est pas à l'équilibre mais dans un déséquilibre dont la dynamique propulse une croissance endogène.

* *

Le cycle du moteur de l'entreprise innovante peut être rompu en deux points: alors ce moteur s'arrête. Supposons que l'entreprise ne parvienne pas à instaurer un monopole temporaire. Alors elle n'a aucune raison d'innover. Cela peut se produire si, par exemple, le régulateur pousse trop loin le combat contre le monopole: il faut donc qu'il contienne sa force et laisse l'entreprise innovante tirer de son effort le surprofit raisonnable.

Supposons que l'entreprise parvienne à prolonger indéfiniment son monopole. Elle est naturellement encline à le faire car cela accroît ses perspectives de profit pour un effort moindre. Le cycle sera alors désamorcé car l'entreprise sera irrésistiblement tentée de se reposer sur ses lauriers.

Pour que le consommateur puisse bénéficier de la croissance que l'innovation alimente, il faut que le monopole temporaire dure assez longtemps, mais pas trop longtemps. Le rythme de la croissance endogène dépend essentiellement de ce délai, tout comme la puissance que dégage le moteur d'une automobile dépend de la position de l'accélérateur.

4.5 Le commerce de la considération

« For senior leaders to work together without debilitating ego clashes, they must strive for mutual respect. If they respect each other, they will find a workable arrangement among themselves. If they do not, the VP responsible for the project must take action because the senior leaders will not remain in a project in which they are not getting the respect they require to succeed. »
(Robert P. Colwell, *The Pentium Chronicles*, [39] p. 16)

L'entreprise type d'aujourd'hui, c'est une entreprise de service : plus de 75 % de la population active travaille dans le secteur tertiaire, où la plupart des agents opérationnels réalisent un travail de bureau (voir le tableau 5.1 page 201). Les fonctions de la première ligne et du *back office* s'enrichissant, il faut parfois les faire tenir par des cadres dont le grade ne correspond plus à une fonction d'encadrement mais à un niveau de compétence.

La sociologie des entreprises les plus modernes se focalise ainsi sur une seule classe moyenne : tout le monde dispose du même poste de travail en réseau, mange à la même cantine, se gare dans le même parking, s'habille de façon semblable ; dans les locaux paysagés, le bureau du directeur ne se distingue des autres (et encore pas toujours) que par sa couleur.

Certes, ce n'est pas une sociologie égalitaire : tout le monde n'a pas accès aux stock-options, les salaires sont inégaux, les pouvoirs de décision aussi. Certes, certaines entreprises croient encore pouvoir faire tenir la première ligne par des personnes mal payées au fort « turn-over » (rares sont par exemple les entreprises qui ont compris l'importance des centres d'appel, voir page 568), mais la frontière sociale entre cols bleus et cols blancs qui se repérait naguère à l'habillement, la coiffure, la peau des mains, l'état de santé, pour ne pas parler du vélo des uns et de l'automobile des autres, ne se retrouve plus dans les entreprises de service modernes.

La diversification des tâches contraste cependant avec l'uniformisation des apparences. La sécurité des systèmes d'information, à elle seule, nécessite plusieurs spécialités dont chacune suppose un travail à plein temps (pare-feux, antivirus, sauvegarde etc.) Par ailleurs l'automatisation, en apportant assistance aux fonctions de première ligne ou de back-office, dégage l'exécutant des tâches routinières et permet de lui demander de prendre des *décisions* qui supposent esprit de synthèse et responsabilité.

Les entreprises ont ainsi recours, dans les tâches d'exploitation comme dans les tâches de conception, à des compétences spécialisées et diversifiées. Or des personnes compétentes ne peuvent travailler de façon efficace que si elles sont insérées dans une organisation où elles puissent se faire comprendre. À quoi servirait par exemple un expert si les décideurs ne comprennent ni son langage, ni son raisonnement ? L'entreprise ne peut former et conserver des personnes compétentes que si elle les *respecte*, des spécialistes ne peuvent coopérer que s'ils respectent les autres spécialités. Certaines directions des ressources humaines y sont de plus en plus attentives¹⁵.

Le respect, ici, n'est rien d'autre que *d'écouter celui qui parle en s'efforçant sincèrement de comprendre ce qu'il veut dire* (« celui qui parle » doit s'entendre au sens large : l'être humain « parle » autant par les gestes, la mimique, les écrits, que par le langage articulé) : on manque de respect à quelqu'un lorsqu'on refuse de l'écouter parce qu'on l'a classé dans une catégorie avec laquelle on croit inutile ou impossible de communiquer.

En principe, nous avons tous de bons sentiments et bonne conscience. Nous ne méprisons personne, nous sommes toute générosité. Pourtant nous cessons souvent d'écouter, nous coupons souvent la parole, si nous n'y prenons pas garde il nous arrive plusieurs fois par jour de manquer de respect à quelqu'un. Ainsi notre pratique est moins respectueuse que notre théorie.

Le respect exige la réciprocité car il devient, à la longue, impossible d'écouter une personne qui elle-même n'écoute pas. Il s'instaure donc dans l'entreprise, et d'autant plus qu'elle sera plus automatisée et que les tâches confiées aux êtres humains seront moins répétitives, un échange du respect, un « commerce de la considération » ; la personne la plus efficace sera celle qui sait amorcer cet échange en acceptant de donner d'abord plus qu'elle ne reçoit. Une direction des ressources humaines efficace sait montrer de la considération aux salariés et aux syndicats, les écouter attentivement et sincèrement, en échange de quoi elle pourra leur demander d'accorder la même considération à la mission de l'entreprise.

Dans certaines organisations hiérarchiques, la personne qui se trouve au sommet exige une obéissance automatique. Elle croit ainsi être respectée alors qu'elle se soustrait au commerce de la considération. Certaines attitudes « respectueuses » équivalent pourtant à la pire des insultes puisqu'elles excluent toute possibilité de dialogue. Certaines religions érigent ainsi leur Dieu à une telle distance qu'aucun dialogue avec lui ne serait possible : ce Dieu, qu'elles croient respecter, elles en font une idole et donc une chose. Et puisque nous sommes arrivés sur le terrain de la théologie, notons que l'injonction « Ἀγαπᾶτε ἀλλήλους » (Jean 13:34), « aimez-vous les uns les autres », résulte d'une traduction de l'araméen au grec. L'araméen étant d'une riche polysémie on aurait pu la traduire par « respectez-vous les uns les autres ».

15. D'après Stiller et Marlowe [126], c'est grâce au soutien de la direction des ressources humaines de Sun que Scott McNealy a pu concevoir le langage Java, qui n'intéressait alors personne dans cette entreprise.

Chapitre 5

À la recherche de la stratégie

On m'a posé un jour la question suivante: « Comment se fait-il qu'il soit si difficile de mettre en place un système d'information, et que ce qui pourrait être fait en trois mois prenne souvent trois ans ou plus? Que la France, jadis à la tête de l'innovation, soit aujourd'hui dans le peloton de queue? Que les décisions semblent dictées plus par la mode que par le raisonnement? »

« C'est, répondis-je, que la majorité des dirigeants de nos grandes entreprises sont non des stratèges, de véritables entrepreneurs, mais des *mondains*. Ils flottent en lévitation au dessus de l'entreprise dont ils ne veulent voir que l'aspect financier. Privée de stratège, celle-ci n'a pas de stratégie. Or il faut une stratégie pour orienter le système d'information, la recherche etc. Ce que je dis là, je le précise, ne s'applique ni aux patrons de PME, qui font ce qu'ils peuvent, ni aux grands patrons du capitalisme familial qui, eux, gèrent attentivement leur patrimoine. »

Je sentis que je contrariais. Les dirigeants remplissent une fonction sacerdotale; les critiquer, cela passe pour du mauvais esprit. Personne ne souhaite favoriser l'anarchie dans l'entreprise. Par ailleurs le mot *mondain* a choqué. On aurait peut-être préféré l'expression « homme de pouvoir » que l'on croit plus flatteuse. Mais l'homme de pouvoir, quand il accapare la légitimité sans pour autant être un stratège, n'est-il pas en effet un mondain dont le principal souci est de gérer sa propre image?

S'il arrive que les propos de cantine soient excessifs on aurait tort de négliger le témoignage qu'ils apportent. Or que dit-on entre collègues à la cantine, dans la plupart des entreprises? Que les dirigeants passent leur temps à se lancer des peaux de banane; qu'ils se neutralisent mutuellement; qu'ils ne recherchent pas l'intérêt de l'entreprise mais seulement le succès de leur propre carrière. Ce comportement-là, c'est celui des *mondains*. Et s'il faut respecter la légitimité des dirigeants, ne doit-on pas savoir distinguer parmi eux le mondain du stratège, comme Saint-Simon (1675-1755) savait distinguer le « plat courtisan » du « fidèle serviteur du Roi »?

Le problème est masqué par l'écart entre langage et comportement. Beaucoup de dirigeants disent avoir besoin de compétences, mais font partir les salariés qui ont atteint l'âge de cinquante-cinq ans ; ils voudraient que les salariés prissent plus d'initiatives, mais sacquent le premier qui ose en manifester ; ils disent le système d'information stratégique, mais ajournent les décisions qui permettraient de le structurer ; ils posent au décideur, mais appliquent la règle « pas de vagues ».

J'ai travaillé dans des ministères et des directions générales d'entreprise. J'ai créé et animé des PME à haute densité de matière grise. J'ai rencontré ainsi quelques stratèges que je respecte et admire mais, je le répète, les stratèges ne sont pas en majorité parmi les dirigeants de nos grandes entreprises, et d'ailleurs dans la lutte pour le pouvoir leur sérieux les handicape.

La plupart des grands pôles de légitimité étant ainsi parasités par des personnes qui n'ont rien à y faire, la paralysie s'étend. Comment enrayer cette épidémie ?

* *

Puisque l'on parle de « stratège », qu'est-ce donc que la « stratégie » ? Ce mot est souvent prononcé avec emphase, sur un ton sentencieux : il désignerait les « choses importantes », celles à propos desquelles il convient de « faire l'important ». Mais dans une entreprise, à vrai dire, tout est important, même et surtout la routine quotidienne : que peut-elle devenir si son réseau cesse de fonctionner ? si elle n'est plus alimentée en matières premières ?

Revenons à l'étymologie : « stratégie » désigne l'ensemble des questions qui concernent le Stratège, ὁ Στρατηγός : le général à la tête de son armée, le gouverneur à la tête d'une province, le consul à la tête de l'État, et par analogie le dirigeant à la tête d'une institution.

Dans une entreprise il est efficace de concentrer le pouvoir de décision légitime, tout comme il est efficace de confier la conduite d'une automobile à une seule personne et non à tous ses occupants simultanément. La fonction stratégique est donc spéciale et importante sans doute, mais non plus importante que les autres car toutes sont également nécessaires, selon la logique du modèle en couches (voir page 42).

Les questions stratégiques sont celles qui méritent l'attention du dirigeant, qui relèvent de sa sphère de responsabilité propre, sur lesquelles sa décision est attendue. Cependant, comme peu de personnes sont appelées à exercer ces fonctions, on en fait un mystère comme s'il s'agissait de magie ou de religion - ce que reflète d'ailleurs le mot « hiérarchie » (cf. la note 1 page 152). Si seule une petite minorité de personnes était autorisée à conduire une voiture, les chauffeurs seraient enviés et admirés car on penserait que la conduite est quelque chose de très difficile (ce qu'elle est en effet, mais comme tout le monde peut y accéder on ne s'en avise pas).

5.1 L'informatique est-elle stratégique?

En mai 2003, la *Harvard Business Review* a publié un article retentissant (Nicholas G. Carr, « IT does'nt matter »¹[32]). Cet article se situe dans la lignée sceptique amorcée par Robert Solow avec son fameux « paradoxe »², suivi par son fidèle Robert Gordon [71], et qui s'est poursuivie, entre autres, avec Thomas Landauer [109].

En réponse la revue a reçu un flot de courrier qui a été publié dans le numéro suivant (« Does IT Matter? »³ *Harvard Business Review*, juin 2003)⁴.

Comme on retrouve dans l'article de Carr un écho de ce qui se dit aujourd'hui dans les entreprises et les SSII, il est utile de l'examiner de près. Beaucoup de gens craignent que l'informatique ne devienne « le textile du XXI^e siècle », qu'elle ne soit incapable de fournir des débouchés alors que le conseil se banalise, que des « usines informatiques off-shore » s'installent, que les DSI sont soumis à un fort « turn-over ».

* *

Carr annonce la fin du règne des TIC et de l'informatique. Celle-ci est devenue, dit-il, une *commodity*, un bien banal que l'on achète sur étagère et auquel on ne pense pas plus qu'à l'air que l'on respire. Cela ne veut pas dire qu'elle ne soit pas importante (une personne privée d'air mourrait en quelques minutes), mais que comme tout le monde y a également accès elle n'est plus un facteur de différenciation et de compétitivité. Certes, l'entreprise qui prendrait du retard en informatique se mettrait en danger car sa fonction de production ne serait pas conforme à l'état de l'art. Mais celle qui prend de l'avance ne peut plus espérer le même avantage concurrentiel que celui dont ont profité naguère FedEx, American Airlines, Mobil Oil, Reuters, eBay, Wal-Mart et autres Dell. L'informatique n'est donc plus, dit Carr, un enjeu stratégique.

En conséquence il recommande de réduire le budget informatique; de n'investir dans des solutions innovantes qu'après que d'autres aient réussi; d'ajourner les investissements pour tirer parti de la baisse des prix; de s'appliquer à gérer les risques et les fragilités plutôt qu'à rechercher des opportunités.

Carr réserve le qualificatif de « stratégique » aux facteurs de compétitivité et de différenciation qui conditionnent la conquête des parts de marché. Limiter les risques, selon lui, ne relève donc pas de la stratégie. Il révèle ainsi qu'il a de celle-ci une conception purement offensive: on peut lui objecter que la stratégie qui négligerait la défensive ferait mal son métier.

1. « Les TIC n'ont aucune importance. »

2. « You can see the computer age everywhere but in the productivity statistics », « on voit des ordinateurs partout, sauf dans les statistiques de productivité » (Robert Solow, *New York Review of Books*, 12 juillet 1987).

3. « Est-ce que les TIC ont de l'importance? »

4. Ces textes sont disponibles sur le site de Yann Gourvenec à l'adresse ygourvenec2.online.fr/webcom/carr

Carr cite des statistiques dont on peut tirer des conclusions opposées aux siennes : « En 1965, selon une étude du service économique du ministère du commerce américain, les entreprises américaines consacraient moins de 5 % de leurs dépenses d'investissement aux TIC. Après l'introduction du PC au début des années 1980, ce pourcentage est monté à 15 %. Au début des années 1990, il a atteint plus de 30 % et à la fin de cette décennie il était de près de 50 %. » ; « le coût de la puissance de traitement a décliné continuellement, de 460 \$ par MIPS en 1978 à 50 \$ par MIPS en 1985, puis à 4 \$ par MIPS en 1995, et cette tendance se poursuit sans faiblir » : ces données témoignent à la fois de l'importance qu'a prise l'informatique dans le capital fixe et d'une baisse de prix qui n'a pu qu'accroître son efficacité à coût égal. Il est banalement vrai que « comme les TIC sont devenues le principal investissement pour la plupart des entreprises, on ne peut plus excuser le gaspillage ni la négligence » ; mais le raisonnement dérape lorsque Carr dit - c'est le moment clé de sa démonstration - « les TIC sont, avant tout, un mécanisme de transport : elles transportent l'information numérisée tout comme les chemins de fer transportent des marchandises, et le réseau électrique transporte le courant. »

Dire que l'informatique est un pur « mécanisme de transport » c'est négliger la modélisation des processus et la définition des concepts opérationnels qui lui donnent ses fondations sémantiques ; c'est oublier qu'avant d'être transportées les données doivent être inscrites dans une mémoire qu'il faut définir et gérer ; qu'elles sont soumises à des traitements qui les transforment ; c'est enfin et surtout compter pour rien l'assistance que l'informatique apporte au travail humain et l'articulation qu'elle permet entre l'être humain et l'automate (voir page 200).

Pour pouvoir tirer argument de la comparaison avec les réseaux de transport (chemin de fer et électricité), il fallait que Carr affirmât qu'« *au fond*, l'informatique *n'est que* du transport ». Mais ce type de raisonnement constitue, en matière de modélisation, une faute majeure : il implique de réduire la nature spécifique de l'objet considéré à celle d'un objet mieux connu et supposé analogue⁵. Or si l'analogie peut être utile dans l'étape heuristique de la pensée elle ne peut pas servir à fonder une démonstration.

Carr fait une deuxième faute lorsqu'il s'appuie sur des statistiques pour prouver que l'informatique n'accroît pas l'efficacité des entreprises. Pour évaluer l'efficacité d'une technique nouvelle il faut en effet considérer non l'ensemble de ses utilisateurs mais seulement les plus habiles d'entre eux, ceux qui ont su trouver la meilleure méthode pour en tirer parti, car c'est cette méthode-là qui s'imposera par la suite à tous. Ce n'est donc pas la statistique qu'il faut utiliser dans ces phases de transition, mais la monographie (Tirole et Lerner [112]).

5. Appliqués au monde de la pensée, ces raisonnements peuvent souligner une synonymie : « *au fond*, la science économique *n'est que* la science de l'efficacité ». Appliqués au monde de la nature (physique ou sociale), par contre, ils procurent de ces inepties que l'on entend au Café du Commerce : « *au fond*, les hommes politiques *ne sont que* des voleurs ».

Certes, il y a actuellement une crise de l'informatique, mais le diagnostic de Carr est-il le bon? Ce qu'il dit s'applique aux réseaux télécoms et, de façon plus générale, aux infrastructures du système d'information (mémoire, puissance, progiciels de base). Par ailleurs il a raison de dire que l'entreprise doit s'efforcer de maîtriser ses dépenses en informatique, se soucier de son degré d'informatisation et des qualités essentielles que son système d'information doit posséder : personne ne peut contredire des recommandations aussi banalement évidentes. Il est vrai aussi qu'elle doit s'appliquer à définir efficacement la frontière de l'externalisation (quelles sont les compétences, les outils que l'on doit conserver dans l'entreprise, quels sont ceux qu'il vaut mieux confier à des fournisseurs? quels sont les programmes spécifiques que l'on doit conserver, et les progiciels que l'on doit se procurer?).

Pendant il ne dit pas qu'il reste beaucoup à faire pour maîtriser le système d'information au plan sémantique (urbanisation, modélisation des processus, qualité des référentiels, administration des données, professionnalisation des maîtrises d'ouvrage, appropriation du système d'information par les métiers etc.). Cette dimension a été souvent négligée alors même que l'entreprise investissait massivement dans l'infrastructure informatique et télécoms. Ce constat est évoqué dans plusieurs des réponses à son article : « Il se peut que les TIC deviennent banales, mais le discernement nécessaire pour tirer parti de leur potentiel ne sera pas réparti de façon aussi égalitaire (...) Tirer bénéfice des TIC requiert une innovation dans les méthodes de travail de l'entreprise » (réponse de John Seely Brown et John Hagel) ; « L'avantage concurrentiel ne vient pas des PC. Il résulte d'une gestion efficace, effectuée par des gens habiles et fortement motivés » (réponse de Paul A. Strassmann) ; « La recherche récente a montré que les méthodes de gestion qui accompagnent l'introduction des TIC coûtent aux entreprises de cinq à dix fois plus que ne coûtent les TIC elles-mêmes » (réponse de Vijay Gurbaxani).

Si l'on admet que la difficulté principale réside dans la sémantique du système d'information, les priorités ne sont plus d'ordre technique⁶ mais d'ordre à la fois philosophique (la sémantique est affaire de concepts, de pertinence, d'adéquation de la pensée à l'action) et sociologique (comme la maîtrise des processus touche à l'organisation, elle pose des problèmes aux personnes comme aux corporations).

Or les progrès de l'entreprise sur les plans philosophique et sociologique sont plus laborieux que les progrès techniques :

- ils supposent un effort collectif de la part de personnes qui n'y avaient pas réfléchi auparavant (le cercle des techniciens aux compétences pointues étant étroit, il est plus facile d'y parvenir à un consensus) ;
- les entreprises ont pris la mauvaise habitude de mépriser la philosophie que les ingénieurs qualifient de « baratin » (Dagognet [42]) ;
- les philosophes eux-mêmes se sont peu intéressés à l'entreprise, si ce

6. La faisabilité technique doit cependant être toujours vérifiée, car même si elle ne représente plus la difficulté principale elle reste une condition nécessaire (donc contraignante) de la réalisation.

n'est pour la dénigrer ou (plus rarement) pour en faire l'apologie, démarches également dogmatiques et qui souffrent des mêmes défauts de méthode que celle de Carr.

Les changements de savoir-faire et de savoir-vivre qu'implique l'informatique nous occuperont pendant la majeure partie du XXI^e siècle. Ils ne se feront pas sans casse car le refus des changements peut être violent. Mais ce point de vue là ne s'est pas exprimé - si ce n'est de façon implicite - dans l'échange de courriers qui a suivi l'article de Carr.

5.2 L'alignement stratégique

Pour dire que le système d'information doit être adapté à la stratégie de l'entreprise on évoque la nécessité de son « alignement stratégique » (Mandel [125]). La stratégie, nous l'avons dit, c'est l'art du stratège; sous un autre rapport on peut dire aussi qu'elle est la poursuite d'un but dans un univers incertain. Cet art, chacun le pratique en petit stratège pour son propre compte lorsqu'il prend les décisions qui conditionneront sa vie personnelle: choix d'une profession, formation d'un couple etc. Dans l'entreprise, les questions de ce type sont celles qui concernent le *positionnement* et l'*organisation*.

Le positionnement désigne la gamme des produits de l'entreprise, les marchés sectoriels ou régionaux qu'elle vise, les fournisseurs qu'elle choisit, les partenariats qu'elle noue, les investissements qu'elle réalise (et, par voie de conséquence, sa politique d'endettement); l'organisation désigne la répartition des responsabilités et des moyens entre les divers pôles de légitimité, le choix des personnes qui vont les diriger, la politique à long terme pour constituer et utiliser un patrimoine de compétences.

La stratégie, étant l'art du dirigeant, se conçoit pour partie dans la solitude mais se vérifie et se précise dans les conseils: on parle de « stratégie de l'entreprise » pour désigner la décision à laquelle le dirigeant s'est arrêté après consultation du comité de direction.

* *

L'expression « alignement stratégique du système d'information » signifie d'abord que le système d'information correspond à la stratégie de l'entreprise, qu'il fournit au personnel les outils permettant de la mettre en œuvre. La relation entre la stratégie et le système d'information serait alors une transcription, le système d'information « tirant les conséquences » de la stratégie (figure 5.1).

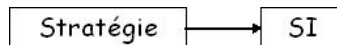


FIG. 5.1 – *Alignement stratégique du système d'information*

Pour que ce schéma fonctionne il faut que la stratégie soit *explicite*. Il ne suffit pas en effet, pour définir un système d'information, de dire de

façon vague que l'on voudrait faire quelque chose : il faut préciser *comment* on entend le faire. Parfois cet exercice d'explicitation fait apparaître des points sur lesquels l'entreprise est non pas volontaire, mais *velléitaire*. Par exemple elle dit vouloir « se mettre à l'e-business » mais rien n'indique à quel genre d'e-business elle entend « se mettre » ; ou bien elle prétend conclure des partenariats mais elle ne fait rien pour assurer l'interopérabilité de son système d'information avec celui de ses partenaires potentiels ; ou encore elle dit vouloir mettre en place un workflow mais sans définir les responsabilités, moyens et pouvoirs de la personne chargée de l'administrer etc.

La précision qu'exige le système d'information contraint l'entreprise soit à renoncer à des projets par trop velléitaires, soit à assumer les implications pratiques de sa volonté stratégique. La réflexion sur le système d'information contribue ainsi à la qualité de l'expression stratégique, ce que l'on peut représenter par une flèche en retour (plus fine toutefois que la première flèche) (figure 5.2).



FIG. 5.2 – Retour du système d'information sur la stratégie

Supposons que nous ayons effectivement défini puis mis en place le système d'information qui correspond à la stratégie, que nous ayons « aligné » le système d'information sur la stratégie, rétroaction comprise. La dynamique ne s'arrête pas là. En effet, la mise en place d'un système d'information ouvre à l'entreprise des possibilités stratégiques qui n'existaient pas auparavant :

- les tickets de facturation émis par un magasin de grande distribution ou les données sur la consommation téléphonique qu'un opérateur télécoms rassemble pour établir la facture peuvent alimenter des études de marketing, des travaux statistiques, qui serviront à fonder une démarche commerciale proactive ;
- les données recueillies par un institut statistique à l'occasion des enquêtes, d'abord exploitées pour produire les tableaux de résultats, peuvent être réutilisées après fusion avec les données d'autres enquêtes ;
- un organisme de sécurité sociale qui recueille les feuilles de soin pour procéder à des remboursements peut utiliser l'information ainsi rassemblée pour alimenter la politique de santé, évaluer la qualité des prescriptions médicales, identifier les fraudeurs.

Le système d'information, d'abord mis au service du positionnement existant, modifie donc le champ du possible et ouvre la perspective d'un nouveau positionnement. Le raisonnement est analogue à celui qui concerne la différenciation des services offerts sur une plate-forme technique : si par exemple le réseau télécoms permet d'offrir le service téléphonique, son infrastructure se prête aussi à l'offre du transport de données, de réseaux privés virtuels, de services à valeur ajoutée etc.

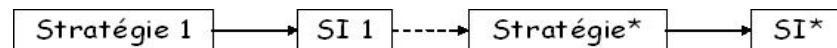
Supposons que l'entreprise tire parti du *patrimoine en information* que représente le système d'information correspondant à la Stratégie 1 ; cela lui permet de définir une Stratégie 2, plus ample que la Stratégie 1 (figure 5.3).

FIG. 5.3 – *Nouvelle stratégie*

Il se peut que déploiement de la Stratégie 2 comporte de nouvelles exigences en termes de système d'information ; notons « SI 1 » le système d'information qui correspond à la Stratégie 1, « SI 2 » celui qui correspond à la Stratégie 2 (figure 5.4).

FIG. 5.4 – *Nouveau système d'information*

Prolongeons le raisonnement : le système d'information 2 va lui-même ouvrir de nouvelles perspectives, ce qui conduira (toujours sous la contrainte de rentabilité) à la Stratégie 3 etc. jusqu'à ce qu'on arrive à un couple stable que nous noterons Stratégie* et SI*. Alors seulement l'alignement stratégique est achevé : la perspective de positionnement que le système d'information procure correspond exactement à la stratégie en place et réciproquement le système d'information en place correspond exactement à la stratégie (figure 5.5).

FIG. 5.5 – *Bouclage de l'alignement stratégique*

Le passage de la Stratégie n à la Stratégie $n + 1$ ne peut bien sûr être envisagé que s'il respecte la contrainte de rentabilité, c'est-à-dire s'il entraîne un accroissement de la valeur de l'entreprise (au sens du cash-flow net anticipé et actualisé). Cependant ce passage n'est pas toujours réalisé, même lorsqu'il accroît la valeur de l'entreprise. En effet il n'est pas facile pour une entreprise de modifier son positionnement, son organisation, ses compétences ; par ailleurs d'autres entreprises peuvent s'opposer à ce changement du positionnement par des voies politiques, en s'appuyant par exemple sur les avis de la CNIL, sur les dispositions légales contre les monopoles ou, si l'entreprise est une administration, sur les textes réglementaires qui délimitent ses missions.

Certains de ces blocages peuvent se justifier (notamment ceux dus au souci de préserver les libertés individuelles) à condition d'évaluer leur justification en regard de la perte d'efficacité qu'ils occasionnent. L'édifice du droit, et en particulier la réglementation qui fixe les missions des adminis-

trations et des établissements publics, datent d'une époque où les possibilités n'étaient pas les mêmes. Il faut oser les mettre à jour mais cela prend du temps. On peut aussi développer dans l'entreprise une compétence en « ingénierie d'affaires » qui aura le sens politique nécessaire pour lever les obstacles externes.

5.3 Du côté des dirigeants

Un « dirigeant », c'est quelqu'un qui occupe dans l'entreprise un pôle de légitimité : soit c'est le pôle suprême (président-directeur général et directeur général, directeur d'administration centrale), soit c'est un pôle spécialisé (DGA, directeur d'une grande direction).

La légitimité est une fonction nécessaire. Elle arbitre entre les projets que l'entreprise produit sans cesse ; elle définit les priorités et oriente les évolutions. Si personne n'était légitime, les affrontements entre équipes et entre personnes seraient sans fin. En les dénouant, l'arbitrage amorce le cycle qui, poussant chaque projet soit vers la réalisation, soit vers l'abandon, dégage l'espace pour de nouveaux projets.

La légitimité a ainsi une fonction mécanique, indépendante de la qualité des décisions : elle fait avancer l'entreprise. Mais il n'est pas indifférent de savoir si celle-ci va vers un précipice, tourne en rond ou se dirige vers un territoire fertile qu'elle pourra mettre en exploitation : si la légitimité est nécessaire, elle n'est pas suffisante.

Nous utilisons ici le mot « stratège » (on pourrait dire aussi « entrepreneur ») pour désigner le dirigeant efficace (voir page 180). Si tous les dirigeants possèdent la légitimité, tous ne sont pas des stratèges (selon une expérience certes limitée mais étayée par de nombreux témoignages, seuls 10 à 20 % des dirigeants mériteraient ce qualificatif). La liturgie qui entoure la légitimité fait d'ailleurs obstacle à la qualité de la stratégie.

* *

Ses pairs confèrent au nouveau dirigeant, lors d'un conseil d'administration, le pouvoir légitime. La cérémonie se réduit à un échange de signatures, à quelques poignées de mains et quelques coupes de champagne mais, tout comme l'onction épiscopale, elle transmet une grâce d'état : le nouveau dirigeant, du seul fait qu'il a été nommé, est censé posséder les compétences nécessaires à sa fonction.

La nomination lui confère effectivement des capacités : il aura le pouvoir de signer des décisions qui engagent l'entreprise, de téléphoner à d'autres dirigeants, de les rencontrer, d'être écouté lorsqu'il parle. Sa parole, sa signature seront nécessaires pour concrétiser les projets et désigner les responsables. Il a ainsi les moyens d'agir. Mais quelles sont les qualités qu'il faut posséder pour être nommé ? Ou, comme le disait naïvement un de mes amis, « que faut-il faire pour devenir dirigeant ? »

Pour que les dirigeants en place aient envie de coopter quelqu'un, il faut que sa compagnie leur soit agréable. Distinction discrète, culture générale, éloquence claire, don de répartie, bonne tenue à table, humour délicat et

sérieux, goût pour les meilleurs vins et cigares, art d'entrer dans les bons réseaux et de s'y maintenir (sortir d'une grande école ou d'un grand corps de fonctionnaires y aide, s'intéresser au rugby ou au golf aussi) : voilà quelques-unes des qualités qui aident à pénétrer l'exquis milieu des dirigeants français. Il est habile de les compléter par l'allégeance à un dirigeant en place.

Ces qualités ne sont pas faciles à acquérir mais on peut les simuler. Un de mes camarades ambitionnait d'entrer au tour extérieur dans le corps de l'inspection des finances. Je l'ai croisé dans les couloirs de Bercy, vieilli, voûté sous le poids imaginaire des dossiers, les pans de la veste battus par un mouvement alternatif des bras qui ramaient l'air. Ayant ainsi montré qu'il possédait l'habitus du haut fonctionnaire, il fut coopté haut la main. Après quoi il rajeunit, sa taille se redressa et ses bras retrouvèrent un mouvement normal.

Vous avez sans doute observé que je ne mentionnais ni la compétence, ni l'expérience parmi les qualités nécessaires : c'est qu'elles ne sont pas absolument indispensables, la grâce d'état y pourvoyant. C'est ainsi que l'on a nommé Michel Bon à France Telecom, Jean-Yves Haberer au Crédit Lyonnais, Jean-Marie Messier à la Générale des Eaux etc. La suite des événements a montré que la grâce d'état pouvait avoir des ratés.

Pour une mission suicide, comme de redresser l'entreprise après un désastre, on préférera tout de même une personne compétente. Mais elle sera jetable : si elle échoue, elle disparaîtra ; si elle réussit, on s'en débarrassera.

* *

Celui qui a les qualités du stratège est peut-être celui qui a le moins envie de devenir un dirigeant : il anticipe les difficultés de cette tâche, le poids des responsabilités qu'elle comporte. Ces difficultés et ce poids, le mondain ne les entrevoit pas ; par contre il est sensible aux privilèges qui accompagnent la fonction de dirigeant. Voiture avec chauffeur, vols en première classe, des collaborateurs pour faire les tâches matérielles, parler sans être jamais contredit, lire l'admiration et la servilité dans les yeux des autres... Étant considéré comme un mâle dominant (ou une femelle dominante) le dirigeant se voit enfin fréquemment proposer des services sexuels : il ne faut ni s'exagérer ce dernier point, ni le passer sous silence⁷.

Un de mes amis, directeur général d'une grande entreprise pendant des décennies, se retrouva sur le sable après que celle-ci eût été achetée par un groupe étranger. Il dut réapprendre à coller les timbres sur les enveloppes, à prendre le métro, à composer des numéros de téléphone, à solliciter des personnes qui ne souhaitaient pas lui parler. Il fut assez sage pour prendre cela avec humour mais ce virage fait sombrer beaucoup d'anciens dirigeants dans la dépression.

On trouve aussi, parmi les aspirants aux fonctions de dirigeant, des pervers qui ne convoitent la légitimité que pour pouvoir faire souffrir les autres, mais il sont rares : un pervers satisfera plus aisément ses penchants dans les

7. Il existe certes des dirigeants que cette offre n'intéresse pas, mais les émules de Louis XIV ou de Catherine II sont nombreux : le pouvoir est un aphrodisiaque.

fonctions de petit chef, moins en vue et plus faciles à conquérir. Parmi les mondains et les pervers peut enfin se glisser un vrai stratège, tout comme il se trouvait de vrais chefs de guerre parmi les courtisans de Versailles. Mais c'est là une coïncidence. Il arrive aussi que l'entreprise, dirigée par un mondain, emploie des personnes qui auraient les qualités du stratège : étant privées des moyens d'action que confère la légitimité, elles doivent ronger leur frein.

* *

Certains disent que l'essentiel pour l'entreprise réside dans la finance. Cette niaiserie les classe tout à côté de ceux qui disent qu'un peintre doit « peindre de beaux tableaux » et un écrivain « écrire des romans intéressants » : de telles phrases font se hausser les épaules du professionnel qui sait devoir se concentrer non sur le résultat final mais sur ses conditions d'émergence, diablement complexes.

Quelle est la place de la finance dans les préoccupations du stratège ? Ou, pour préciser la question, qu'attend-il du directeur financier ?

1) D'abord, que celui-ci gère la trésorerie de l'entreprise, ses créances et sa dette : il doit placer les liquidités, couvrir les risques (change, contrepartie, fluctuation du cours des matières premières), minimiser le coût de l'endettement et le poids de la fiscalité, toutes opérations hautement techniques.

2) Ensuite, qu'il conforte la *crédibilité* de l'entreprise, son aptitude à obtenir du crédit. Toute entreprise est en effet structurellement endettée auprès des actionnaires, des banques et des fournisseurs. Si les créanciers exigent un remboursement immédiat ou s'ils refusent de renouveler les prêts elle peut être mise en faillite. Elle doit donc entretenir la confiance de ceux que l'on appelle, par abus de langage, « les marchés ».

Certes ces missions sont importantes, mais pas au point que l'on puisse y réduire la stratégie : l'arbitrage entre les projets que les concepteurs produisent sans cesse, le maintien de la fonction de production à l'état de l'art, l'utilisation opportune des techniques nouvelles supposent des connaissances et une réflexion qui ne relèvent pas de la seule finance.

La crédibilité financière de l'entreprise ne se construit d'ailleurs ni uniquement, ni même principalement sur les paramètres financiers. Les créanciers, pour savoir s'ils peuvent lui faire confiance, examinent la qualité de ses produits, la solidité de sa part de marché, sa réactivité face aux évolutions techniques ou réglementaires, sa capacité à s'emparer d'un avantage concurrentiel en innovant.

Il est assez naturel qu'un dirigeant, s'il a été coopté parce qu'il émettait l'image convenable, voie dans l'image de l'entreprise le facteur principal de la crédibilité. Il accordera tous ses soins à la communication, à la présentation du bilan, fût-ce en sacrifiant des actifs précieux pour faire apparaître un résultat séduisant mais fugace. Il se détournera des conditions pratiques de son fonctionnement, de son évolution, pour monter de ces acquisitions qui accaparent l'attention des journalistes et des actionnaires. L'image de l'entreprise, détachée de son socle économique, devient alors un artefact

médiatique qui peut monter très haut avant que les actionnaires, apercevant du vide sous le cours de l'action, ne soient pris de vertige.

* *

En paraphrasant Pascal⁸, on peut dire que *la vraie finance se moque de la finance* : la santé financière de l'entreprise résulte d'une stratégie qui agit, en priorité, sur autre chose que la finance. Le stratège est comme le jardinier qui, plutôt que de tirer sur les plantes, bine le sol, le fume, l'irrigue et le sarcle.

Dans les années 1990 plusieurs très grandes entreprises sont mortes ou ont failli mourir parce que leurs dirigeants avaient donné la priorité à la finance et à la communication ; plutôt que de créer de la richesse, de l'utilité pour les consommateurs, ils entendaient « créer de la valeur », cette expression convenue signifiant dans leur bouche « faire monter le cours de l'action ». Or le marché boursier, volatil par nature, connaît des oscillations sans rapport avec la santé de l'entreprise : le prendre pour boussole est suicidaire.

On pouvait espérer qu'après les désastres des années 1990 (France Telecom, Vivendi, Crédit Lyonnais etc.), provoqués par des mondains jouant au dirigeant, nous reviendrions à une conception raisonnable de l'entreprise. Hélas ! On entend encore des journalistes, des économistes, des professeurs répéter les mêmes niaiseries sur la priorité financière de la stratégie. Les étudiants, nourris de ces viandes creuses, en redemandent : dépourvus d'expérience pratique et prisonniers de la médiatisation de la société, ils confondent volontiers l'image et la réalité.

5.4 La stratégie retrouvée

Le mot « stratège » évoque les praticiens de l'art de la guerre : Xénophon (426-355) (*L'Anabase*), T. E. Lawrence (1888-1935) (*The Seven Pillars of Wisdom*), le général Leclerc (1902-1947).

Dans l'entreprise, on peut citer Bob Crandall (1935-) qui dirigea American Airlines de 1980 à 1998. Il disait « Je suis un homme à idées, et quand on est un stratège (un *leader*) on peut tester ses idées. » Crandall est à l'origine de SABRE, qui fut en 1973 le premier système de réservation électronique ; de l'organisation du réseau en « Hub and Spokes », du « Yield Management », du programme « Frequent Flyer », du « b-scale » etc. Ces innovations, qui s'appuyaient toutes sur une utilisation intensive et novatrice du système d'information, ont permis à American Airlines de dominer pendant un temps ses concurrents. Prises dans leur ensemble, elles ont transformé le transport aérien.

Chaque lundi, Crandall réunissait son équipe. La réunion durait toute la journée. Les divers aspects de l'entreprise, représentés par des nombres, alimentaient des graphiques préparés pendant le week-end. Rapports et études

8. « La vraie morale se moque de la morale » (Blaise Pascal (1623-1662), *Pensées*, 24).

circulaient autour de la table. Celui qui se trouvait incapable de répondre à une question portant sur son domaine était accablé de sarcasmes. Crandall était colérique, mais on pouvait le calmer en lui opposant un argument logique (Petzinger [159]).

Citons aussi Herbert Kelleher (1931-), créateur de Southwest, compagnie *low cost* devenue le plus important transporteur aérien aux États-Unis. Kelleher, qui s'est lui aussi concentré sur la physique de l'entreprise, a conçu un type de réseau (navettes point à point) entièrement différent de celui que Crandall a organisé, mais tout aussi efficace.

* *

Le stratège concentre son attention non pas sur le milieu social des dirigeants, mais sur l'entreprise et son environnement. En mûrissant une synthèse, il fournit à l'entreprise une *orientation*, un « sens ». Cette activité, qui dégage les priorités de la polyphonie des métiers et des accidents externes, lui interdit de s'enfermer dans une spécialité : il est à l'écoute des experts qui l'assistent et alimentent le processus de décision.

L'horizon temporel du stratège va de quelques mois à quelques années ; ce n'est pas celui de la gestion quotidienne et l'expression « pilotage stratégique » est un oxymore.

Parfois le stratège doit décider vite, mais cela s'inscrit sur la toile de fond d'une connaissance lentement mûrie, qui soutient une appréhension intuitive et presque physique de l'entreprise. Le « coup d'œil », vertu qui permet de décider avec justesse sous la pression de l'urgence et du danger, est la qualité du stratège parvenu au sommet de son art⁹.

Dans l'entreprise, le stratège observe en priorité (1) la fonction de production, d'où se déduit la fonction de coût ; (2) les besoins des clients, suivis par le marketing et schématisés par la segmentation.

La connaissance de la fonction de production permet de dimensionner les ressources (compétences et effectifs, équipements, partenariats), de définir la nature et le calendrier des investissements, de se maintenir à l'état de l'art, de réduire le coût de production, de générer un surprofit par des innovations de procédé. La connaissance des besoins permet de diversifier la production, de la « packager » et la tarifier de façon à satisfaire les divers segments de clientèle, de définir les services qui accompagnent sa commercialisation et sa distribution, de générer un surprofit par des innovations de produit.

À cette concentration sur l'entreprise, le stratège ajoute une vigilance périscopique sur les initiatives de la concurrence, les évolutions des techniques et de la réglementation, la crédibilité de l'entreprise (figure 5.6). Il n'ignore certes pas la finance mais, encore une fois, elle n'est pas sa préoc-

9. « Rien de plus juste que le coup d'œil de M. de Luxembourg, rien de plus brillant, de plus avisé, de plus prévoyant que lui devant les ennemis, ou un jour de bataille, avec une audace, une flatterie, et en même temps un sang-froid qui lui laissait tout voir et tout prévoir au milieu du plus grand feu et du danger du succès le plus imminent ; et c'était là où il était grand. Pour le reste, la paresse même. » (Saint-Simon (1675-1755), *Mémoires*, Gallimard 1983, vol. I p. 207. (*Nota Bene* : La langue de Saint-Simon n'est pas la nôtre ; il faut traduire « flatterie » par « assurance » et « danger du succès » par « risque »).

cupation unique ni même principale sauf bien sûr si l'entreprise est au bord de la faillite.

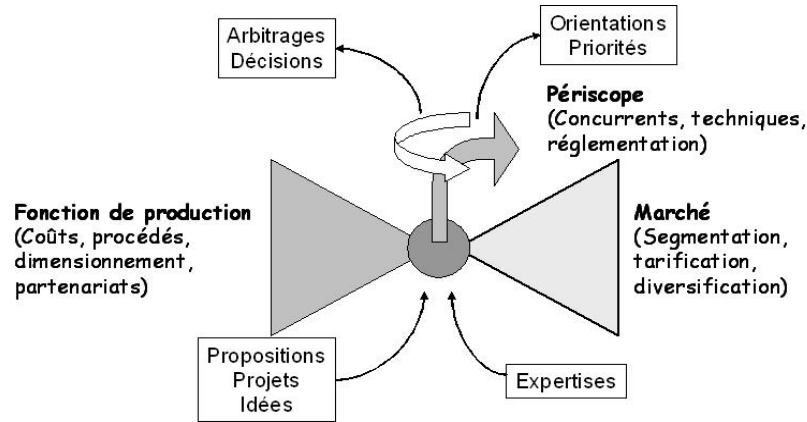


FIG. 5.6 – *Portrait du stratège*

Pour savoir si une entreprise est ou non dirigée par un stratège, examinez la qualité de son système d'information et, en particulier, le tableau de bord du comité de direction (voir page 379). Cela donne une première indication.

Si le tableau de bord est sobre, lisible et pertinent en regard de la physique de l'entreprise, la réponse sera vraisemblablement positive : seul un stratège est capable d'imposer aux directions la production d'un tel tableau de bord, de surmonter les réticences des détenteurs de l'information.

Souvent il n'existe pas de tableau de bord ou bien - ce qui revient au même - il en existe plusieurs mutuellement contradictoires : alors le comité de direction flotte en lévitation au-dessus d'une entreprise dont il ignore la physique et qui n'a donc ni stratège, ni stratégie.

* *

Le rôle du système d'information dans la stratégie est souvent mal compris ou dévoyé. Je vais l'illustrer en citant des anecdotes toutes authentiques et que je crois représentatives.

Écartons auparavant deux malentendus :

1) Certaines personnes croient que les anecdotes, étant *anecdotiques*, n'apportent aucun enseignement qui vaille : il faudrait leur préférer les statistiques, les enquêtes, qui seules fourniraient une information bien pondérée.

Eh bien des statistiques existent, on peut les trouver à la page 453, et elles montrent que dans la plupart des cas les projets informatiques échouent ou connaissent d'importants dépassements en délai et en budget. Une fois que l'on a vu la statistique, reste à chercher l'explication d'une ingénierie aussi défectueuse : et ici les anecdotes sont précieuses, car elles permettent d'orienter l'intuition. Il y a d'ailleurs quelque chose d'étrange dans le mépris envers les anecdotes : sauf mensonge, elles rapportent des *faits* indéniables et

c'est à partir du constat de tels faits que peut s'amorcer l'enquête statistique elle-même, qui le complétera par des ordres de grandeur, des proportions et des corrélations.

2) Certaines personnes croient que celui qui ne fait pas taire son jugement devant la fonction des dirigeants et, plus généralement, devant les institutions, témoigne d'un mauvais esprit. Or c'est tout le contraire : si l'on respecte la fonction des uns, la mission des autres, on doit s'interdire toute complaisance envers leurs éventuels dévoiements. Dans le cas particulier du système d'information il se trouve que le dévoiement est la règle et la réussite l'exception : c'est là sans doute un fait navrant, mais dont répétons-le la statistique témoigne de façon irréfutable.

Je ne voudrais d'ailleurs pas que l'on crût que j'éprouve de l'hostilité envers les dirigeants. Quand je rencontre un dirigeant efficace, un entrepreneur, un vrai stratège, j'éprouve envers lui de la gratitude car par son efficacité il contribue au bien-être de la société tout entière. J'essaie aussi de comprendre comment il s'y prend et ce n'est pas facile : la décision juste résulte d'une adéquation du jugement à la situation, d'une adhésion à la « propension des choses » (Jullien [93]) que l'on ne saurait entièrement expliciter¹⁰ et qui semblent dépasser les facultés strictement rationnelles sur lesquelles elles s'appuient pourtant.

Les mécanismes mentaux de l'erreur sont beaucoup plus faciles à comprendre et à décrire que ceux de la justesse ; c'est pourquoi je leur accorde autant de place, abordant ainsi par l'extérieur, et en quelque sorte par leur contour négatif, les qualités positives du stratège.

* *

Au milieu des années 90 un président-directeur général convaincu de l'importance stratégique du système d'information demanda à un de mes amis de concevoir une démarche ambitieuse. Mon ami fit le tour de l'entreprise et revint avec un constat : il fallait d'abord organiser les maîtrises d'ouvrage pour équilibrer le poids de la direction informatique, remplacer les terminaux passifs par des PC en réseau, faire la chasse aux doubles saisies, mettre en place la messagerie et la documentation électronique, créer un référentiel, monter un tableau de bord, ouvrir un site pour faire du commerce sur la Toile, enfin reconcevoir certaines applications en utilisant l'architecture client/serveur (elle était alors à la mode) et un langage à objets.

Le président-directeur général donna son accord mais il fallait, pour que les choses puissent se faire, qu'il donnât des consignes au directeur général. Il n'en fit rien. Mon ami consacra plusieurs mois à des négociations stériles. Il finit enfin par comprendre ce qui se passait. Alors il dit ceci au président-directeur général :

« Quand je parle de poste de travail, de référentiel, de tableau de bord, de maîtrise d'ouvrage, de site Web, tout cela vous paraît mesquin, banal, car cela n'est pas à la hauteur de la Stratégie que vous ambitionnez. Pourtant en

10. Talleyrand disait de Bonaparte, premier consul, « il comprend tout ce qu'il voit et il devine ce qu'il ne voit pas » (Orioux [149]).

matière de système d'information la stratégie se construit non par des décisions spectaculaires mais plutôt par une succession cohérente, persévérante de dispositions modestes, terre-à-terre, qui élargissent progressivement les possibilités offertes à l'entreprise.

« Vous dites le système d'information stratégique mais vous ne faites rien. Vous êtes comme un homme amoureux d'une femme qu'il trouve belle, fine, intelligente et dont il se fait une très haute idée. Un jour cette dame lui dit : "Je sais que tu m'aimes et toi aussi, tu me plais ; viens t'allonger tout contre moi". Mais il reste paralysé : ne sachant comment surmonter l'écart entre l'idéal et la réalité, l'action lui est impossible. Vous êtes l'amoureux transi du système d'information ».

Le président-directeur général pâlit et ne répondit rien. Comme il ressemblait un peu à Louis XIV je dirai, dans le style de Saint-Simon, que mon ami avait mal fait sa cour.

* *

Un autre de mes amis est chargé par un directeur régional de l'équipement de concevoir le tableau de bord de la direction. Il fait son travail mais entre temps ce directeur est nommé ailleurs. Le tableau de bord est présenté à son successeur.

Mon ami explique : « chaque semaine, vous aurez ce tableau-ci ; chaque mois, ce tableau-là. »

« Holà, dit le directeur, *je saurai tout ça?* »

« Oui », répond mon ami.

« Mais alors il faudra que je prenne des *décisions?* »

« Eh bien oui : en quelque sorte, c'est fait pour. »

« Alors ça, non, je ne veux pas de ce système ! »

Le tableau de bord n'a pas été mis en place. L'économiste dira « c'est parce que la fonction d'utilité du second directeur n'était pas la même que celle du premier », mais il reste à expliquer pourquoi il en était ainsi.

Mettons-nous à la place de ce directeur. S'il ne sait pas ce qui se passe, il ne pourra rien faire mais on ne pourra rien lui reprocher. Il mènera sa vie de notable : grand bureau, secrétaire diligente, subordonnés respectueux, voiture, chauffeur. Sans tableau de bord, il n'y aura pas d'information donc ni responsabilité ni soucis. Il gagnera convenablement sa vie jusqu'au jour béni de la retraite.

* *

Le chef de projet avait été mal choisi ; l'équipe se transformait en petit royaume indépendant au sein de l'entreprise ; les prévisions de coût et de délai croissaient sans cesse.

Le maître d'ouvrage délégué auprès du directeur général proposa une tactique à celui-ci : « Le chef de projet est à contre-emploi : il est sérieux, dévoué, mais quoiqu'on puisse lui dire il n'a jamais compris le but du projet. Il croit de son devoir de "satisfaire la demande des utilisateurs" sans y faire le moindre tri, sûr moyen de se planter. Son équipe vise à conforter sa propre

pérennité et, pour rendre impossible sa dissolution en fin de projet, elle se prépare à faire en double le travail des autres services.

« On ne peut rien reprocher de grave à ces personnes qui sont honnêtes au fond, et agissent plus par instinct et par ignorance que par calcul. Mais le projet va à l'échec.

« Je vous suggère de tirer parti des difficultés budgétaires pour leur dire : "Je suis désolé mais cette année je n'ai pas assez d'argent. Il faut que j'arrête un projet et ce ne peut être que le vôtre. Merci pour vos efforts." Puis vous faites ce qu'il faut pour qu'ils ne se sentent ni humiliés, ni désavoués : vous les félicitez, vous les décorez, mais enfin vous les renvoyez dans les directions régionales d'où vous les aviez fait venir. L'année prochaine, vous trouverez les ressources budgétaires nécessaires et relancerez le projet. Vous le ferez redémarrer en nommant un chef de projet compétent, avec une autre équipe. »

Le directeur général soupira et répondit : « Je ne peux pas faire ça à Jean-Claude (prénom du directeur dont le projet dépendait), c'est le seul grand projet de sa direction. Il ne s'en remettrait pas. » C'était là un bel exemple de compromis managérial (voir page 566). La suite des événements a confirmé le diagnostic pessimiste.

* *

Une de mes amies était responsable du système d'information d'une grande fédération affiliée au MEDEF. Partant du désordre habituel, elle construit les fondations d'un système d'information adéquat à l'action de cette fédération : référentiel ; bases de données ; portail Web ; information sur les entreprises de la branche, permettant de savoir qui fait quoi, qui a quelles relations avec la fédération, de calibrer les actions de communication etc. Je passe les détails.

Elle reçoit les félicitations du directeur général : « ce que vous préparez est stratégique ». Son travail suscite des jalousies, mais qu'importe quand on est soutenue par le directeur général ?

Elle prend un congé de maternité, et pendant ce congé tout se déglingue. Le contrat de réalisation est ôté au premier fournisseur pour être mis entre les mains d'un autre, qui doit tout reprendre depuis le début. L'organisation est bouleversée à tel point que plus rien ne peut fonctionner.

Mon amie argumente en vain par téléphone. Elle finit par envoyer un courrier pour expliquer les problèmes et signaler les blocages. Ce courrier est jugé insultant pour un des directeurs : dès son retour de congé elle est licenciée pour faute lourde, après quoi elle subira une dépression.

J'ai dit à cette amie : « Lorsque le directeur général a dit "ce que vous faites est stratégique", c'est comme s'il t'avait dit "vous êtes condamnée à mort". Pendant la conversation, certes, tu éveillais sa matière grise ; cela lui était agréable et ses félicitations étaient sans doute sincères. Mais le soir, chez lui, il se disait "qu'est-ce qu'elle est en train de me préparer, cette bonne femme ? Tout cela va me compliquer la vie, il va falloir se débarrasser d'elle". Il t'a encouragée d'autant plus volontiers qu'il savait pouvoir démolir ton travail à l'occasion de ton congé de maternité : il lui suffisait d'attendre. »

Un ami était maître d'ouvrage délégué auprès du directeur général d'une entreprise dont l'avenir était suspendu à un grand projet. Ce projet allait mal : la SSII ne tenait pas ses engagements, les délais s'allongeaient, la dépense croissait, la correction des anomalies ne convergeait pas, l'« effet tunnel » jouait à plein. Le directeur de projet, étant du côté de la maîtrise d'ouvrage, n'avait aucun contrôle sur le budget ni sur la vérification du service fait qui étaient jalousement gardés par la DSI. Il affichait un optimisme de commande : « ça finira bien par tomber en marche », disait-il. Le DSI, de son côté, gourmandait ses troupes inquiètes : « vous êtes des enfants, leur dit-il un jour textuellement. Le logiciel, c'est toujours merdique. Il ne faut pas exiger la perfection ».

Les années passaient. Après une phase initiale quelque peu hystérique et marquée par des conflits de personnes (il y avait du pouvoir et de la gloire à ramasser), la maîtrise d'œuvre avait perdu le moral. La maîtrise d'ouvrage modifiait sa demande tantôt pour de bonnes raisons réglementaires, tantôt par caprice. La réflexion stratégique était bloquée en l'attente du produit, censé tout résoudre.

Mon ami proposa au directeur général une manœuvre de sauvetage : mettre au vert les responsables pendant une semaine, réviser les intentions stratégiques qui fondaient le projet, préparer une position de repli. « Je ne peux pas faire ça, répondit le directeur général, car le projet s'arrêterait ». « Vous êtes, lui dit mon ami, comme le commandant d'un paquebot qui n'oserait pas faire manœuvrer les canots de sauvetage de peur que cela ne fasse couler le navire. Si votre entreprise est aussi fragile que cela, il faudrait en tirer les conséquences ! »

Quelques années plus tard, le projet sera arrêté après avoir coûté de l'ordre d'une centaine de millions d'euros.

* *

Certains systèmes d'information avancent ainsi :

- un jeune ingénieur plein de bonne volonté conçoit un projet solide, cohérent, raisonnable, conforme à l'état de l'art ;
- il commence à le mettre en œuvre avec la bénédiction de la direction : « ce que vous faites est stratégique » ;
- des décisions qui compromettent le projet sont cependant prises ; il regimbe, on dit qu'il a mauvais caractère, qu'il est prétentieux ;
- la situation devient absurde, l'action impossible, les relations se tendent ;
- finalement il part ou il est viré : sentiment d'échec, dépression, divorce parfois ;
- une partie du projet reste érigée comme un chantier inachevé au milieu du système d'information ;
- un jeune ingénieur plein de bonne volonté conçoit... (retour au premier alinéa).

Le système d'information est ainsi construit, au prix de la destruction des personnes qui en ont été successivement chargées, par juxtaposition de réalisations partielles. Il est et reste incohérent.

Certains réussissent malgré tout à faire progresser le système d'information. Ce sont des personnes d'âge mûr qui ont compris comment cela fonctionne. Elles savent qu'il faut surtout éviter le compliment : « ce que vous faites est stratégique ». Se tenant à l'affût des occasions, elles font les choses sans le dire. Comme elles ont la bonhomie et la rugosité des ours on n'ose pas les contrarier.

J'ai connu de ces « ours ». Ils ne sont ni fortement diplômés, ni haut placés dans la hiérarchie. Ce sont des cadres moyens qui travaillent dans de petits bureaux et n'ont pas d'assistante. Ils sont maître d'ouvrage opérationnel d'une des grandes applications de l'entreprise, sur laquelle ils se sont spécialisés et qu'ils connaissent dans ses moindres recoins. C'est à eux que s'adressent les utilisateurs en cas de problème : « je n'arrive pas à sortir tel état », « la réglementation vient de changer, comment adapter l'appli » etc.

Ils ne parlent jamais de « méthodologie » ni de « démarche qualité » (ces expressions leur font plutôt hausser les épaules), mais tout ce qu'ils disent est exact, tout ce qu'ils font est judicieux. Ils s'entendent à merveille avec les informaticiens, ravis d'avoir un client à l'esprit clair. Les dirigeants les plus lucides disent « je ne sais pas ce que nous ferions si Untel passait sous un autobus, il sait tout et il est le seul à tout savoir ». Et en effet l'ours est un solitaire qui déteste partager son travail et n'aime pas écrire.

Grâce à ces ours, et sans tambour ni trompette, le référentiel se met en place, la qualité des données s'accroît, les solutions deviennent convenables, les utilisateurs sont satisfaits. *Mais cela prend beaucoup de temps*, et des incohérences subsistent : chaque ours se focalisant sur une seule application, aucun d'eux ne voit clairement les exigences transverses et ils sont mal placés pour évaluer les priorités.

* *

J'ai rencontré aussi des entreprises évolutives, efficaces, rapides, économes de leurs moyens : mais elles sont rares. Dans la plupart des entreprises il ne convient pour progresser ni de parler de stratégie, ni même de communiquer, mais d'agir dans la durée de façon à créer une situation irréversible et nouvelle.

Est-ce parce que nous sommes en France ? Quand on voit comment ont été traitées les personnes d'Enron et de Worldcom qui avaient détecté et dénoncé les fraudes, ou quand on lit le rapport d'audit sur le système d'information du FBI (page 498), ou quand on voit les statistiques que publie le Standish Group (page 453), on voit que le même problème se rencontre aux États-Unis. Il se rencontre sans doute dans tous les pays.

Dans la plupart des entreprises, dans la plupart des pays on attend le jour où les dirigeants sauront se représenter de façon raisonnable les apports et les limites de l'informatique.

5.5 Restaurer le mot « Informatique »

Nous avons pris l'habitude d'associer au mot « informatique » une connotation purement technique : il désigne la plate-forme des mémoires, processeurs, réseaux et logiciels. Nous appelons « informaticiens » les spécialistes qui gèrent cette plate-forme. Trop souvent l'entreprise les considère comme des « techniciens » enfermés dans leur spécialité et dont elle déplore « l'autisme ».

Pour désigner le côté sémantique, le référentiel, les fonctionnalités fournies aux utilisateurs, nous disons « système d'information ». Certes il est vain de s'insurger contre l'usage une fois qu'il s'est bien installé. Je voudrais cependant dire pourquoi cet usage me semble non pertinent, et comment j'en suis venu - à ma grande surprise - à réévaluer le mot « informatique ».

* *

Le concept de « système d'information » est apparu dans les années 1970 lorsqu'on a cherché à surmonter les inconvénients qui résultaient de la superposition non coordonnée d'applications différentes. Il a été introduit en France par Jacques Mélése [136], influencé par Herbert Simon [186].

Selon la théorie qui a alors prévalu, une organisation complexe doit être analysée en distinguant son utilisation des langages (système d'information), ses règles de conduite et de comportement (système de décision) et les procédés qu'elle met en œuvre (système de production). Ces trois systèmes possèdent chacun sa propre structure et communiquent entre eux (figure 5.7).

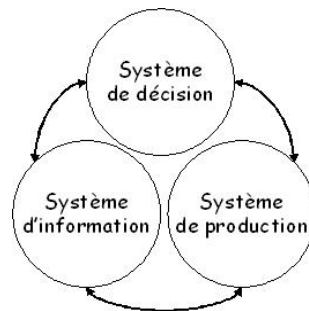


FIG. 5.7 – *Les trois systèmes de l'entreprise*

Cette théorie a permis de mettre de l'ordre dans les applications informatiques en les articulant toutes à un même référentiel et en organisant les échanges entre bases de données - dans la plupart des entreprises, cette mise en ordre est loin d'être achevée.

Cependant cette théorie ne rend pas compte de la situation présente. D'une part, l'informatique équipe maintenant la partie automatique des processus de production et de gestion, et s'articule ainsi de façon étroite avec le travail humain : elle est « près du corps » dans toutes les activités de l'entreprise ; d'autre part, l'introduction de l'informatique de communica-

tion, puis son articulation avec l'informatique de calcul, ont encore resserré la relation entre l'informatique et l'organisation. Désormais, comme le dit Yves Tabourier, « on ne peut ni changer l'organisation d'une entreprise sans changer son système d'information, ni changer son système d'information sans changer son organisation ».

Il était commode de modéliser séparément les trois domaines « production », « décision » et « information » car il est impossible de modéliser ni même de penser quoique ce soit en postulant que tout est dans tout et réciproquement. Mais comme aujourd'hui on ne peut plus penser l'organisation séparément du système d'information, il faut utiliser un autre découpage que celui qui les sépare.

L'*urbanisation* du système d'information (voir page 433) a fourni une première réponse : elle découpe dans l'entreprise des « domaines de production » aux contours généralement proches de (mais pas toujours identiques à) ceux des grandes directions ; elle identifie les processus à l'œuvre dans ou entre ces domaines, puis les activités qui concourent à ces processus, enfin les tâches que doit remplir l'automate pour assister efficacement les agents chargés de ces activités (aides à la saisie, au traitement, au classement, à la consultation, à la communication).

La frontière ne passe plus alors entre « production », « décision » et « information », mais entre les divers domaines de l'entreprise, chargés chacun d'une production spécifique et faiblement couplés entre eux (mais couplés quand même : les échanges entre domaines sont l'un des points délicats de l'urbanisation). À l'intérieur de chaque domaine, et même de chaque processus, on retrouve les trois pôles du découpage précédent : la production (rôle opérationnel du processus) et la décision (gestion et animation du processus), toutes deux entrelacées à l'information.

Le découpage des domaines et des processus par l'urbaniste comporte, comme tout découpage, une part d'arbitraire : il est impossible de définir le découpage optimal. S'agissant d'une tâche pratique, la recherche d'un optimum serait d'ailleurs sans doute superflue : l'examen de l'entreprise¹¹ fait apparaître l'enchaînement des activités et la solidarité des processus ; les frontières des directions donnent des indications sur celles des domaines¹² et, à quelques détails près, des urbanistes différents aboutiraient au même découpage.

L'urbanisme fournit donc une nouvelle articulation : celle des domaines, processus et activités. Cependant, et fondamentalement, la démarche urbanistique met en scène une autre articulation, plus profonde : celle qui met en relation les êtres humains et l'automate, et qui définit l'assistance que celui-ci apporte à ceux-là.

Comment distinguer en effet, dans chaque processus, les tâches qu'il convient de confier à l'automate, et celles qu'il faut réserver à l'être humain ? Où placer la frontière de l'automatisation ?

11. Cet examen doit être critique, car il ne convient pas de perpétuer les défauts éventuels des processus existants.

12. Toutefois ces frontières peuvent résulter de compromis boiteux entre sphères de pouvoir : leur examen doit être lui aussi critique.

Quand l'informatique outille une usine automatisée, c'est l'ensemble de l'usine qui constitue l'automate ; mais lorsqu'elle outille un processus ou interviennent des êtres humains, il faut la définir de telle sorte que le couple que constituent l'automate et l'être humain soit globalement efficace, ce qui suppose (a) que chacune des parties puisse délivrer sa propre valeur ajoutée, (b) que l'articulation de ces deux valeurs ajoutées dégage une synergie.

Comment désigner les deux parties qui s'articulent ainsi ? L'être humain que nous considérons ici n'est pas la personne dans son individualité inefable, mais la *personne au travail*, insérée dans l'entreprise où elle exprime sa compétence et la conjugue à d'autres selon les canaux de l'organisation : il s'agit donc de l'« être humain organisé », EHO, ensemble de professionnels qui coopèrent.

D'autre part, l'automate auquel l'EHO s'articule est *essentiellement programmable* (il est potentiellement apte à faire tout ce qu'un programme peut prescrire) et le réseau lui confère l'ubiquité (les services qu'il rend sont indépendants de la localisation géographique de l'utilisateur comme de celle des serveurs ; cet automate se situe donc dans un espace purement logique) : c'est l'« automate programmable doué d'ubiquité », APU.

Quelles sont les valeurs ajoutées que sont susceptibles d'apporter l'APU d'une part, l'EHO de l'autre ? Pour peu qu'on lui ait donné des instructions parfaitement explicites, l'APU offre la fidélité de la mémoire et la puissance du traitement, conjuguées à une patience et une obéissance illimitées. Sa tâche sera donc de classer, retrouver, transcrire, calculer, recopier, transmettre. Les tâches répétitives fatiguent l'EHO, mais il est supérieur à l'APU quand il s'agit de comprendre, expliquer, décider et concevoir. Contrairement à l'APU, il est capable d'interpréter une information incomplète ou suggestive¹³.

L'articulation entre l'EHO et l'APU peut dégager une synergie si chacun se consacre à la tâche qu'il fait mieux que l'autre et si les interfaces qui les relient sont convenablement définies. La définition de leur articulation requiert une analyse des conditions d'exécution du processus. L'efficacité résultera d'une automatisation « raisonnable » (voir « éloge du semi-désordre » page 99).

Cette efficacité représente un enjeu important pour notre économie et notre société, comme l'indique le calcul sommaire du tableau 5.1¹⁴. La part de la masse salariale « consacrée à l'informatique » (part du temps de travail des salariés passée devant l'écran-clavier) est aujourd'hui de 30 à 40 %. Elle va croître encore : on peut estimer qu'au début des années 2010 plus de la moitié de la masse salariale rémunérera un temps de travail passé devant l'écran-clavier, donc dans l'univers du système d'information.

La qualité de l'APU, et celle de son appropriation par l'EHO, sont dès maintenant des enjeux essentiels pour les entreprises même si celles-ci n'ont pas encore perçu toutes les implications de ce phénomène.

13. Il ne faut pas pour autant mépriser l'APU ! Le travail qu'exécute le pilote automatique d'un avion de ligne épuiserait vite un pilote humain.

14. Sources : ligne 1 : [213], p. 37 ; lignes 2, 3 et 4 : Acadys, www.acadys.com ; la colonne relative aux années 2010 est une extrapolation.

	Années 1980	Années 1990	Années 2000	Années 2010
(1) Part du Tertiaire dans l'emploi	55 %	65 %	75 %	80 %
(2) % des salariés du Tertiaire équipés	5 %	35 %	70 %	90 %
(3) Dont % du temps de travail sur ordinateur	15 %	35 %	60 %	75 %
(4)=(2)*(3) Poids de l'informatique dans le Tertiaire	0,8 %	12,3 %	42 %	67,5 %
(5)=(1)*(4) Poids de l'informatique dans le temps de travail	0,4 %	8 %	31,5 %	54 %

TAB. 5.1 – Poids de l'informatique dans le temps de travail

* *

Philippe Dreyfus a inventé le mot « informatique » en 1962 pour baptiser une société qu'il venait de créer, la SIA (« Société d'Informatique Appliquée »). Ce mot, qui n'avait heureusement pas été déposé, a été adopté en 1967 par l'Académie française pour désigner la « science du traitement de l'information ». Puis il fut adopté par de nombreux pays¹⁵.

La terminaison « tique » renvoie à « automatique », donc au processus de traitement des données. La première partie du mot indique qu'il s'agit d'automatiser le traitement de l'*information* (et non des *données*, car Philippe Dreyfus aurait alors dû construire « datamatique ») : or une donnée ne peut devenir une information que quand un être humain l'interprète¹⁶ (voir page 36). La conjonction des mots « information » et « automatique » suggère la coopération entre l'être humain et l'automate, à condition bien sûr de distinguer « donnée » et « information ». En créant « informatique », et peut-être sans en avoir clairement conscience, Philippe Dreyfus nous a fourni ainsi un mot dont l'étymologie convient parfaitement pour désigner l'articulation entre l'EHO et l'APU.

La qualité du mot « informatique » paraît donc supérieure à celle de l'expression *computer science* qu'il traduit¹⁷ : il accumule en effet un ensemble

15. On dit « Informatik » en allemand, « informática » en espagnol et en portugais, « informatica » en italien, « информатика » en russe.

16. Certains soutiennent que les mécanismes comportant une rétroaction (pilote automatique d'un avion, thermostat d'un chauffage central etc.) « décident », « agissent » et à cette fin reçoivent une information. Nous préférons dire qu'ils sont programmés lors de l'introduction manuelle des paramètres (cap et altitude pour le pilote automatique, température pour le thermostat), et qu'ensuite ils soulagent l'opérateur humain en traitant de façon continue les données que leur fournissent des capteurs.

17. Ce n'est pas l'opinion de Donald Knuth : « Computer science is known as “informatics” in French, German, and several other languages, but American researchers have been reluctant to embrace that term because it seems to place undue emphasis on the stuff that computers manipulate rather than on the processes of manipulation themselves » (Knuth [104] p. 3) : or le processus de manipulation semble correctement représenté par

de notions plus riche que celle de « calculateur », de *computer*, même si dans *computer science* apparaît le mot « science ».

Il est vrai que l'on a souvent donné au mot « informatique » le sens de « traitement des données », l'interprétation de celles-ci (la « sémantique ») relevant alors du système d'information. Mais cette restriction fait perdre la richesse que comporte l'étymologie d'« informatique », la coopération et la tension qu'elle implique entre l'être humain et l'automate.

Le terme « télématique¹⁸ » suggère la synergie entre l'informatique et les télécommunications ; il a été traduit en anglais (*telematics*) et utilisé en France pour qualifier le programme Minitel lancé par la direction générale des télécommunications en 1981. Mais comme presque tous les ordinateurs travaillent aujourd'hui en réseau l'automate a désormais acquis l'ubiquité : l'apport du terme « télématique » s'est ainsi résorbé dans le terme « informatique ».

L'expression « système d'information » oriente l'intuition vers un « système », donc vers une structure¹⁹, alors même que l'on entend désigner un être qui, lié de façon organique à l'entreprise, évolue avec elle. Le mot « informatique », grâce à la tension qui le sous-tend, convient sans doute mieux pour représenter cette dynamique.

Il faut, pour rendre au mot « informatique » la rigueur et la vigueur que comporte son étymologie, s'affranchir des connotations qui s'y sont accolées. L'« informatique », dans l'entreprise, est trop souvent considérée comme un centre de coût ; les « informaticiens », comme une corporation sur la défensive. Si la plupart des entreprises se sont informatisées, on ne peut pas dire qu'elles aient toutes donné à l'informatique, dans leur culture et leurs priorités, la place que celle-ci mérite. La mauvaise qualité du vocabulaire est corrélative de cette confusion des priorités. Si l'on se libère des habitudes, on découvre dans ce mot un potentiel sémantique plus fort que celui auquel il a été réduit par l'usage.

Nommer un concept n'est pas une opération neutre : le nom signale le concept à l'attention de celui qui réfléchit, et il en est le vecteur d'échange. Ainsi en *nommant* l'articulation de l'être humain et de l'automate, nous l'avons *signalée* : nous en avons fait pour l'entreprise à la fois un phénomène, un sujet de préoccupation et un objectif.

Le phénomène a existé bien sûr dès que des entreprises ont utilisé des ordinateurs, mais il n'a été d'abord ni reconnu ni donc pensé. Les recherches, dans la ligne des réflexions de Turing (Turing [206]), se sont d'abord focalisées sur l'intelligence de l'ordinateur, ce qui équivalait à postuler une identité de nature entre l'ordinateur et le cerveau humain²⁰. Dès lors il devenait im-

la terminaison « automatique ». Knuth aurait pour sa part préféré « algorithmics », qui correspond à ses propres priorités (op. cit. p. 88).

18. Introduit en 1978 par Nora et Minc [135].

19. On risque de tomber alors dans le même travers que les structuralistes qui, des travaux de Saussure, n'ont voulu retenir que la « structure » en négligeant ce que Saussure avait dit sur l'évolution de la langue, sur la façon dont la structure se forme et se transforme (Saussure [179]).

20. Cette hypothèse est encore caressée par des penseurs ou auteurs de science fiction (Truong [205]).

possible de penser leur relation : on ne peut pas articuler l'identique avec lui-même.

Il est naturel que l'exploration se soit engagée sur cette piste car tout inventeur croit trouver la clé du monde dans sa découverte. Cette démarche fut d'ailleurs féconde : la métaphore biologique a aidé à concevoir les fonctions de l'ordinateur et il était nécessaire, dans les années 1950, de combattre la position prétendument humaniste ou religieuse, en fait sentimentale et pétrie de préjugés, de ceux qui estiment que pour promouvoir l'homme il faut ignorer ou éviter l'automate. Pour s'affranchir de ce sentimentalisme il aurait fallu fonder l'humanisme en raison ; dans l'attente d'une telle fondation, l'agacement que provoquent les bons sentiments ou une foi religieuse invoquée à contre-temps a poussé des esprits fermes et courageux dans une impasse.

Dès 1957 cependant von Neumann [147] a compris qu'il était pertinent de postuler non pas l'identité du cerveau et de l'automate, mais leur différence et leur articulation. Sa mort précoce l'empêcha de donner l'écho qu'elle méritait à cette conception à laquelle il attachait la plus grande importance.

* *

Au plan théorique, la réflexion sur l'articulation entre l'EHO et l'APU incite à explorer leurs propriétés, chacun servant de complément et de miroir à l'autre. La rigueur conceptuelle explicite que demandent les bases de données et les référentiels, la diffusion des commentaires que permet la documentation électronique, les outils de classement et de recherche qui l'accompagnent comme elles accompagnent la messagerie, la maîtrise de la qualité permise par les workflows, l'articulation de la parole et du système d'information à travers les centres d'appel etc. modifient en effet les conditions de la mise en œuvre des facultés intellectuelles tout comme, en d'autres temps, le firent l'écriture, la lecture, puis l'imprimerie. La compréhension de ces phénomènes, de leurs conséquences et implications, demande des recherches à peine entamées.

L'automate est d'ailleurs pour le cerveau un utile complément. « Les mathématiques considèrent des descriptions déclaratives (qu'est-ce que c'est), alors qu'en informatique on considère des descriptions impératives (comment ça marche) »²¹ (Abelson et Sussman [197] p. 22). Le raisonnement mathématique peut être assisté par l'automate qui assume le rôle de calculateur, explore des simulations et permet des démonstrations là où le nombre des configurations à considérer est élevé, comme pour le théorème des quatre couleurs (Dowek [50]). Bien plus : le recours au modèle de l'automate a permis d'établir des résultats de métamathématique comme le théorème de Gödel, alors que par ailleurs la pédagogie des mathématiques a dû corriger les excès de formalisme qu'avait encouragés la confusion entre le cerveau et l'automate²².

21. « In mathematics we are usually concerned with declarative (what is) descriptions, whereas in computer science we are usually concerned with imperative (how to) descriptions. »

22. « C'est en croyant s'inspirer de Bourbaki que certains mathématiciens zélés ont in-

Au plan pratique, l'accent sur l'articulation entre l'EHO et l'APU, la restauration du mot « informatique », changent le regard que l'on porte sur le système d'information. On l'avait souvent focalisé sur le développement des produits informatiques ou sur l'« implémentation » des progiciels au détriment de la compréhension des processus de l'entreprise et de l'assistance que l'ordinateur peut leur apporter²³. On a ainsi fait de l'informatique un corps étranger dont la greffe sur l'entreprise présente un caractère artificiel et suscite le rejet. Cependant les progrès en cours (urbanisation, modélisation, langage UML, architecture SOA etc.) militent pour que l'on place cette articulation au centre de la réflexion.

Par ailleurs, l'appropriation de l'automate par l'être humain soulève de nombreuses questions de savoir-faire et de savoir-vivre : en témoignent le foisonnement des spécialités dans la profession des informaticiens, et aussi les questions que pose le bon usage de l'automate et que s'efforce par exemple de résoudre la « netiquette » : comment utiliser convenablement la messagerie (voir page 387)? Un serveur Web? Un système d'aide à la décision(voir page 299)? Comment mettre en place un référentiel (voir page 265)?

La relation entre l'EHO et l'APU est illustrée par la différence entre le langage naturel et le langage conceptuel. Le langage naturel est enrichi de connotations qui facilitent son interprétation au prix d'une imprécision. Par contre le langage conceptuel des mathématiques, de la science, de la technique, est dépourvu de connotations : il est précis, donc bien adapté à l'action qui exige une parfaite précision, mais cela le rend difficile à comprendre et donc impropre à la communication. Lorsque ce langage est utilisé par des êtres humains, ceux-ci se permettent, comme le dit Bourbaki, des « abus de langage » qui allègent les développements. Un programme informatique, tout comme le plan d'une machine ou d'un immeuble, doit par contre être parfaitement explicite et ne tolère aucun abus de langage.

Les missions de l'informatique dans l'entreprise se sont élargies à la façon d'un fleuve qui parvient à son delta : en s'assimilant les réseaux, l'informatique s'est dotée de l'ubiquité ; en absorbant la bureautique communicante, elle s'est assimilée les textes en langage naturel et les commentaires nécessaires à l'interprétation des données ; en équipant non seulement les fonctions administratives de l'entreprise mais l'ensemble des processus de travail, elle a généralisé le travail assisté par ordinateur.

Cette extension a été permise par la baisse du coût de l'informatique ainsi que par la diversification des logiciels. Si, en 1962, le mot « informatique »

introduit les prétendues mathématiques modernes dans l'enseignement secondaire... On a introduit un grand nombre de termes abstraits à la place de mots concrets que tous les jeunes pouvaient aisément comprendre. Pierre Samuel a raillé cette attitude en parlant des "hyperaxiomatiseurs en mal de généralisation". Les ravages causés par l'introduction des dites mathématiques modernes furent une catastrophe internationale de grande envergure, mais plus particulièrement française. Une génération de jeunes Français a été sacrifiée du point de vue de l'apprentissage des mathématiques. » (Laurent Schwartz, cité dans www.apmep.asso.fr/BV442SCH.html)

23. Cela se traduit par des travers que la théorie recommande d'éviter mais qui renaissent sans cesse : mauvaise connaissance des utilisateurs, mauvaise appropriation du système d'information par les dirigeants, spécifications inflationnistes, projets trop nombreux et trop fréquents etc.

recouvrait une réalité essentiellement technique et encore assez pauvre²⁴, il peut aujourd'hui revendiquer toute la profondeur de son étymologie.

Cette extension, nous la devons à des informaticiens et pourtant elle s'est parfois faite contre leur corporation, qui n'a bien accueilli ni les réseaux, ni la bureautique, ni l'extension de l'informatique aux processus de travail. C'est que la population des informaticiens est très composite. La sociologie des chercheurs, des concepteurs, des pionniers, n'est pas la même que celle des techniciens en place, soucieux avant tout de défendre leurs avantages acquis, ni que celle des bureaucrates qui, amateurs de pouvoir, sont habiles à se l'approprier (on ne saurait d'ailleurs classer les personnes selon les frontières de ces trois sociologies : une même personne peut, de façon incohérente mais très humaine, se comporter tantôt en hardi pionnier, tantôt en suppôt frileux de la corporation, tantôt en homme de pouvoir).

Ainsi l'informatique est aujourd'hui située à un carrefour. Son image technicienne et autiste ne correspond plus à la réalité. Si les dirigeants lui reconnaissent un caractère stratégique, ils sont loin d'en tirer les conclusions pratiques. Elle a pénétré toutes les activités de l'entreprise dont elle est à la fois l'appareil sensoriel et le système nerveux. Ses réalisations s'empilent comme des couches géologiques correspondant chacune à une phase de son évolution. Sa sociologie, qui s'est elle aussi forgée au cours de l'évolution, conjugue des traits parfois opposés. Une telle diversité engendre la confusion des idées, des images, des habitudes. Pour échapper à cette confusion il faut trancher le nœud gordien et revenir à une définition claire et simple.

Si nous étions assez souples pour réviser notre vocabulaire, nous abandonnerions l'expression « système d'information » et la remplacerions par « informatique » en conférant à ce mot le sens que nous venons de décrire. Ce que nous appelons aujourd'hui « informatique », nous l'appellerions désormais « plate-forme technique de l'informatique » ou « plate-forme technique » tout court.

Mais une telle évolution du langage est peu vraisemblable tant l'usage est impérieux. Lorsque j'ai tenté de trouver un terme pour désigner l'articulation de l'EHO et de l'APU, j'ai été contrarié de retrouver le mot « informatique » car il se présentait à mon imagination avec toutes ses connotations techniciennes. Mais lorsque je les ai nettoyées pour restaurer son sens étymologique, j'ai vu apparaître sa puissance sémantique. Ce mot magnifique est riche de potentialités négligées. Il est apte à désigner cette articulation de l'être humain et de l'automate qui sera, durant les décennies prochaines, le grand enjeu pour nos entreprises, voire pour notre civilisation - comme le furent, et le sont encore, l'articulation de l'être humain avec l'écriture ou avec le moteur. J'utilise bien sûr l'expression « système d'information » lorsque je parle avec d'autres personnes mais dans mon langage intérieur le rayonnement du mot « informatique » s'est imposé : c'est ce qui explique le titre du présent ouvrage.

24. Chez les chercheurs, par contre, nombre de notions essentielles étaient acquises ; mais il faut un délai pour que les idées des chercheurs passent dans la pratique des entreprises. Les premières applications de l'informatique, tout utiles qu'elles soient, ne visaient qu'à outiller des tâches de gestion (paie, comptabilité, gestion des stocks etc.).

Chapitre 6

Aspects philosophiques

L'évolution des systèmes d'information rencontre souvent des blocages. On entend alors prononcer des phrases d'une remarquable absurdité : des personnes habituellement raisonnables semblent, lorsqu'il s'agit d'informatiser l'entreprise, avoir perdu leur bon sens.

C'est que l'on touche à des frontières fermement tracées dans chaque tête, quoique toujours de façon implicite, et qui séparent ce qui est réel de ce qui n'est qu'apparent, ce qui est important de ce qui est négligeable.

La construction d'un système d'information évoque et réveille les séparations que ces frontières définissent. Il est donc naturel que sa modélisation provoque des réactions de souffrance, de désarroi, et suscite des discussions d'autant plus violentes qu'elles opposent des personnes également maladroites sur le terrain de la philosophie mais qui ne savent comment quitter l'oreiller de leurs évidences familières.

Les philosophes de métier sont eux-mêmes ici de peu de secours. La plupart d'entre eux sont des professeurs. Qu'ils méprisent ou révèrent l'entreprise, ils n'ont jamais vu de près comment elle fonctionne : cela se voit bien lorsqu'ils sont appelés à gérer une université.

Il faudra pourtant puiser dans leurs travaux pour démêler l'écheveau des évidences, des échelles de valeur qui entrent en conflit dans l'entreprise et que la discussion sur le système d'information ravive comme si l'on versait de l'acide sur une plaie.

* *

Lucien Sfez [183], par exemple, nomme « surcode » le fait de croiser divers codes pour représenter un même aspect du monde réel. Il dit attendre du surcodage une « déviance » susceptible de « dynamiser les sociétés ». Son discours illustre l'enthousiasme qui saisit le philosophe lorsqu'il redécouvre, en toute naïveté, des notions depuis longtemps familières à nous autres praticiens.

1) Les statisticiens savent qu'il existe *a priori* une infinité de façons différentes, toutes également correctes au plan formel, de coder des activités économiques, produits, métiers etc. Le respect des règles formelles (cohé-

rence, complétude etc.) ne suffit pas à définir un code : il doit répondre aussi et surtout au critère de *pertinence*, d'adéquation à l'action que l'on vise (Guibert, Laganier et Volle [77]). On doit coder selon ce que l'on a *l'intention de faire*. Il faudra par ailleurs souvent, pour définir entièrement un code, faire aussi quelques choix de pure convention.

Deux institutions qui n'ont pas visé les mêmes actions, ou qui n'ont pas les mêmes conventions, coderont ainsi les mêmes choses de façon différente. Pour que l'une puisse réutiliser les données codées par l'autre il faudra définir des tables de traduction qui seront toujours approximatives.

2) Les « tris croisés » sont familiers aux utilisateurs des bases de données. Lorsque vous considérez la répartition d'une population selon un code, par exemple la répartition de la population française par région à une date donnée, ce « tri à plat » fournit un tableau à une dimension ; si vous considérez la répartition de cette population à la fois par région et par classe d'âge vous produisez un tableau à deux dimensions, comportant plusieurs lignes et colonnes. L'analyse de ce tableau donnera des indications sur la corrélation qui existe, dans cette population et à cette date, entre les classifications par région et par classe d'âge - autrement dit, entre les deux codes que vous avez croisés. « Croiser les codages » ajoute donc, à la connaissance de la répartition de la population selon chaque code, celle de la corrélation entre ces deux codes dans cette population-là. Le constat d'une telle corrélation amorce l'inférence qui conduit aux hypothèses causales et à la théorie.

3) Enfin, le « modèle en couches » familier aux informaticiens et aux spécialistes des télécoms (voir page 42) permet de modéliser des objets (réseau, ordinateur etc.) dont le fonctionnement articule plusieurs logiques (ou « protocoles ») différentes, toutes également nécessaires, et communiquant par des « interfaces ». Le modèle en couches est une innovation philosophique d'une grande portée mais son origine technique constitue, aux yeux de certains philosophes, une tare peut-être irrémédiable.

* *

Les tables de traduction, le croisement des codages, le modèle en couches, sont des « procédés de pensée » (voir page 229) qu'il est bon de savoir utiliser non seulement dans les systèmes d'information, mais dans la vie courante.

Ainsi, lorsqu'on est confronté à une réalité dont la représentation comporte plusieurs dimensions, il est bon de les croiser deux à deux : cela fait apparaître des corrélations qui parfois surprennent et incitent à la réflexion. De même, lorsqu'on est confronté à plusieurs logiques jouant simultanément chacune avec ses contraintes propres et sa temporalité, les séparer en les articulant permet de simuler mentalement leur interaction et d'éviter des erreurs (comme, par exemple, celle si répandue qui consiste à croire que l'informatique se résume à la numérisation alors que celle-ci n'intervient que dans les couches basses de l'ordinateur).

Ces procédés aident l'intelligence de celui qui les utilise mais ils n'ont rien de « révolutionnaire », rien qui puisse suffire à « dynamiser les sociétés ». En effet la traduction, le croisement ou l'articulation de plusieurs codes, *c'est encore un code* : lorsque par exemple on croise les caractères « région » et

« classe d'âge », on produit le code bidimensionnel « région, classe d'âge ». Il faut donc chercher le « surcode » ailleurs que dans ces opérations. Lucien Sfez l'a-t-il entrevu? Il semble que non, mais il se peut que cela se dissimule dans une de ses phrases les plus obscures.

* *

Nous rencontrerons le « surcode » non dans le croisement de divers codes, mais dans la connaissance des *conditions de production* du code.

Si le but du codage est d'outiller l'action, la première exigence est celle de la pertinence. Les règles formelles, certes impératives, sont celles que les pédagogues commentent le plus volontiers car elles ne prêtent pas à discussion ; leur respect n'est pas toutefois pour le praticien le point le plus difficile.

En effet au moment où l'on définit le code, l'action que celui-ci doit outiller n'est pas en cours, elle n'est qu'anticipée. Dans l'écart entre l'intention et l'action effective se glisse l'incertitude sur les circonstances exactes de l'action future.

La pertinence d'un code, ainsi que celle des modèles qui reposent sur lui, peut être alors aussi instable que ne l'est l'équilibre économique lorsque les agents introduisent, dans leurs choix présents, l'anticipation d'un futur essentiellement incertain. Elle peut être altérée par les rigidités qui résultent des erreurs d'anticipation¹. Or trop souvent on oublie que le code en usage fut, un jour, construit. Devenu aussi solide et rigide qu'un bâtiment ou qu'une institution, il arrive qu'il survive à l'évaporation de sa pertinence.

Revenir aux conditions de la production du code, c'est prendre conscience des intentions auxquelles ont obéi ses concepteurs, lui rendre la fraîcheur qu'il avait entre leurs mains, lui conférer enfin la souplesse nécessaire pour gérer la *dynamique de la pertinence*, qui relie l'intention à une action placée dans son contexte. Cette dynamique peut être, tout comme une dynamique économique, aveugle ou maîtrisée. Lorsque nous disons que le système d'information vise à élucider les processus de production de l'entreprise et que les salariés doivent se l'approprier, c'est cette maîtrise-là que nous ambitionnons.

* *

Considérer le code comme un outil fait par et pour la main de l'ouvrier et qu'il faut savoir modifier à l'occasion, c'est tout simplement du réalisme. Ce n'est pas plaider pour une « remise en question permanente », car il ne convient pas de changer chaque jour d'outil. Ce n'est pas non plus se comporter en « déviant » préparant la « Révolution ».

Pourquoi Lucien Sfez a-t-il utilisé un vocabulaire aussi emphatique, et pour désigner de surcroît une conception plutôt pauvre du surcode? C'est parce que beaucoup de personnes considèrent les codes comme des phénomènes naturels et confondent les institutions avec la nature. Lorsqu'on

1. Prendre en compte les anticipations a conduit les économistes, dans le sillage de Keynes, à construire une « théorie du déséquilibre » (Grandmont [73]).

s'efforce de les détromper on rencontre de tels obstacles qu'il est tentant de croire qu'une fois ces illusions surmontées, eh bien la Révolution aura été accomplie.

Mais il n'en est rien. Supposons les illusions supprimées : cela rend sans doute les discussions plus faciles mais il reste encore à élaborer la pertinence des codes, à gérer leur dynamique, et ce n'est pas un mince travail. D'ailleurs s'il est bon pour progresser dans l'efficacité de maîtriser la relation entre action et codage, cela ne garantit pas que l'action en question soit « bonne » du point de vue de l'éthique comme de la qualité des rapports entre personnes. Être plus efficace, mieux comprendre ce que l'on fait, commettre moins d'erreurs, c'est souhaitable mais ce n'est pas la Révolution, si du moins l'on estime que celle-ci doit concerner non seulement l'efficacité mais aussi les valeurs qui orientent l'action.

On peut faire des observations analogues devant les analyses d'Edgar Morin [142]. Ce qu'il appelle « contradiction » et « complexité », c'est l'articulation de diverses logiques dans un même phénomène, articulation dont rend compte le modèle en couches que les ingénieurs savent maîtriser. Mais cette articulation est simple, puisqu'elle est pensable ! Mieux vaut réserver le terme « complexité » à l'opacité de tout existant devant la pensée. Celle-ci ne peut pas élaborer la représentation complète d'un être réel, mais seulement une représentation partielle à finalité pratique, ce qui peut suffire à l'action. Là encore se pose le problème de la pertinence, et aussi celui de la qualité de l'action elle-même.

* *

Beaucoup d'auteurs présentent leurs concepts avec l'enthousiasme d'un pasteur lancé en plein sermon. Avec le « surcode », la « logique du flou » (Davis et Meyer [133]), la « complexité » (Legendre [110]), les « systèmes » (Le Moigne [138] et Lévy [122]), disent-ils, vous changerez le monde !

Il y a souvent du vrai dans leurs analyses et ils ont raison de crier fort. Le système technique mécanisé (Gille [66]), qui a martelé notre culture aux XIX^e et XX^e siècles, a rigidifié la pensée comme les institutions. Il n'est pas facile de s'en libérer. Il faut donc aider beaucoup de personnes, fussent-elles intelligentes, à sortir du sommeil dogmatique si agréable dont Kant lui-même a eu besoin d'être réveillé par Hume².

Mais lorsqu'on a commencé à comprendre certaines choses sur les systèmes d'information, les techniques, les institutions, les codages, les valeurs etc. il reste à organiser sa pensée, à identifier les principes, à conforter les raisonnements. Pour concentrer son énergie, le volcan brûlant du désir doit alors se couvrir de glace : après avoir démoli l'illusion on a besoin, pour reconstruire une pensée pertinente, de rigueur, de précision et de claire sim-

2. *Ich gestehe frei: die Erinnerung des David Hume war eben dasjenige, was mir vor vielen Jahren zuerst den dogmatischen Schlummer unterbrach, und meinen Untersuchungen im Felde der spekulativen Philosophie eine ganz andre Richtung gab* : « Je le reconnais volontiers : c'est le souvenir de David Hume qui, voici plusieurs années, a interrompu mon sommeil dogmatique et donné à mes recherches en philosophie spéculative une orientation complètement différente. » (Emmanuel Kant (1724-1804), *Prolegomena* 1783).

plicité. Les suggestions aussi vagues que puissantes que portent des termes comme « flou », « complexe », « système » ne suffisent plus et deviennent même des obstacles. Pour progresser, il faudra au mieux les redéfinir, au pis les rejeter, tout comme il faut démonter un échafaudage pour habiter la maison qu'il a aidé à construire.

6.1 Articuler la pensée et l'action

À chaque instant nous faisons jouer la charnière qui relie notre pensée et notre action. Mais plutôt que de se tenir à cloche-pied sur la crête qui les sépare en les articulant, notre réflexion préfère souvent dévaler la pente soit vers la pensée, soit vers l'action.

Si les penseurs militaires (Desportes [47]) ont été attentifs à cette charnière cruciale, les philosophes restent, à quelques exceptions près³, accaparés par le versant de la pensée et le commun des mortels par celui de l'action. Pourtant rien n'est plus nécessaire que de penser notre action et que d'agir de façon réfléchie ; mais notre formation nous en a éloignés, tout comme nous en éloigne la culture que diffusent les médias.

Au xx^e siècle la science a en effet rejeté des modèles qu'elle avait auparavant adoptés (le déterminisme, l'équivalence entre vérité et démontrabilité⁴) pour en proposer d'autres dont la compréhension suppose une bonne maîtrise des mathématiques (espace non euclidien de la relativité générale, ondes de probabilité de la mécanique quantique, théorème de Gödel). Devant ces théories d'accès escarpé, devant les apparents paradoxes qu'elles recèlent, l'intellect est resté comme stupéfait.

Si l'on demande à des ingénieurs d'aujourd'hui, passés par le moule des grandes écoles, ce que signifient les mots « réalité » et « causalité », nombre d'entre eux répondront que ces mots n'ont aucun sens. En pratique cependant, et c'est heureux, ils discernent ce qui est réel de ce qui est imaginaire et ils raisonnent sur des causalités, fussent-elles hypothétiques⁵. Mais cet écart entre leur pratique et leur pensée n'est-il pas dommageable ?

L'écart entre la façon dont nous nous autorisons à penser explicitement et la façon dont, en pratique, nous agissons et pensons n'est pas innocent. Chacun restant libre de réfléchir à part soi, cet écart ne nous interdit sans doute pas totalement de définir les objets de notre action, de les modéliser, de les maîtriser par l'intellect ; mais il constitue un obstacle presque insurmontable à la communication, au partage et à la circulation de la réflexion. Or une réflexion qui reste individuelle aura tôt fait de se faner et de se dis-

3. Parmi les exceptions, citons Maurice Blondel [16] et Edmund Husserl [87]. On trouve aussi une réflexion sur l'action chez Confucius et Mencius.

4. Le déterminisme est contredit à l'échelle subatomique par des phénomènes probabilistes, à l'échelle macroscopique par des phénomènes chaotiques ; l'équivalence entre vérité et démontrabilité est contredite par le théorème de Gödel.

5. De même ils utiliseront la mécanique newtonienne dans les échelles courantes d'espace et de temps, la mécanique quantique et la relativité générale ne s'imposant l'une qu'à l'échelle de l'atome (donc dans les transistors et les circuits intégrés), l'autre qu'à celle du cosmos, deux échelles que l'expérience quotidienne rencontre rarement.

soudre. Il en résulte de graves dommages pour l'équilibre des personnes, la clarté des enjeux de société, l'efficacité des entreprises.

Comment pourraient-elles être « souples », « évolutives » si elles estiment « trop intellectuel » d'explicitier la charnière entre la pensée et l'action, de mettre en discussion leurs concepts, composants, processus, modèles etc., si elles refusent de *penser* les référentiels qui fondent la représentation des êtres (clients, fournisseurs, partenaires, techniques, produits, salariés etc.) avec lesquels elles entendent faire évoluer leur relation ?

À l'occasion des partenariats ou des fusions elles découvrent, toujours avec le même étonnement douloureux, des écarts conceptuels qu'il leur est difficile de résorber. Ne devraient-elles pas tirer la leçon d'une surprise qui se renouvelle si souvent ? Osons secouer la timidité nous emprisonnent une formation superficielle et des médias qui cultivent la confusion entre réel et image. Osons recourir aux notions de « réalité » et de « causalité » : cela assainira nos entreprises et, en tout premier, leur système d'information.

* *

La charnière qui relie la pensée à l'action joue à l'intérieur de chacun : chacun pense, agit en fonction de ce qu'il pense, réfléchit pour préparer ses actions futures. Elle joue aussi à l'intérieur de chaque entreprise, où elle est enrichie et compliquée par la division du travail ainsi que par les frontières entre spécialités, entre domaines de l'organisation. Elle joue encore à l'intérieur de la société qui, considérée dans son ensemble, pense et agit collectivement. Un même modèle peut donc *mutatis mutandis* être utilisé pour la décrire, qu'on la considère dans la personne, dans l'entreprise ou dans la société, chacune de ces entités pouvant servir de métaphore à l'autre : on peut dire qu'une entreprise « réfléchit » ou « comprend », qu'une personne « gère », « investit » et « produit ».

Pour voir comment s'articulent la pensée et l'action nous commencerons par la personne : comme chacun pense et agit il suffit de s'examiner soi-même, de se prendre pour terrain d'expérimentation, pour voir comment cela fonctionne. Cela procurera une métaphore facile à comprendre et que nous pourrions ensuite transposer à l'entreprise.

Nous allons démonter la charnière, identifier ses éléments, ensuite seulement reconstituer sa dynamique. C'est ainsi que l'on fait pour comprendre le moteur à explosion : on examine séparément le fonctionnement des pistons, soupapes et bougies avant de se représenter leur interaction. Cette analogie mécanique contrariera ceux qui estiment que ni la pensée ni l'action ne se réduisent à un mécanisme. Ils ont raison, car à la source comme à l'horizon de notre action et de notre pensée se trouvent une volonté et des valeurs qu'un mécanisme ne décrit pas. Toutefois ce n'est pas de ces valeurs que nous entendons parler ici ni de cette volonté, mais seulement de la façon dont elles sont mises en œuvre : et dans l'examen d'une mise en œuvre l'analogie mécanique n'est pas déplacée.

En analysant un fonctionnement qui semble simple, nous procéderons comme les physiologistes qui, pour analyser l'influx nerveux, décrivent des phénomènes électriques et chimiques dans les neurones et à leurs synapses.

Nous appellerons « monde de la nature » la nature physique, biologique, sociale et humaine qui, nous entourant, agit sur nous et subit notre action ; nous le distinguons du « monde de la pensée » de nos idées, réflexions, représentations, raisonnements et décisions.

Le « monde de la nature » produit des *événements* : quelqu'un nous parle ; nous recevons une lettre ; un client entre dans notre magasin ; alors que nous conduisons notre voiture, nous arrivons à un feu rouge ; une machine tombe en panne ; à la chasse, un sanglier débouche devant nous etc. L'événement suscite en nous une *expérience* qui est son image dans le monde de la pensée⁶.

À l'état brut, l'expérience est informe : le tout petit enfant perçoit des zones colorées, des mouvements et des sons parmi lesquels il ne distingue que le visage et la voix de sa mère. Le développement mental nous a dotés d'un *discernement* plus ou moins délié : dans une forêt celui qui ne connaît pas la botanique voit « des arbres », celui qui en sait un peu plus voit « des bouleaux, des mélèzes, des hêtres ». Le discernement délimite notre *perception* de l'expérience : nous ne percevons que ce que nous savons discerner, pour le reste nous sommes comme de petits enfants.

Le discernement s'appuie sur une *grille conceptuelle*, sur les concepts dont nous nous sommes équipés pour classer et séparer les éléments de l'expérience. Apprendre à piloter un avion, c'est d'abord acquérir les concepts qui conviennent au pilotage. Celui qui n'a pas appris à piloter ne sait discerner ni les signaux que donne le tableau de bord, ni ce qui apparaît par les vitres du cockpit, ni les commandes.

La grille conceptuelle fournit à l'action le langage selon lequel elle peut être pensée, concertée et décidée. Sans un tel langage, l'action est soit impossible, soit désordonnée. La grille conceptuelle est essentiellement *pratique* (orientée vers l'action) et non, comme on le croit le plus souvent, intellectuelle (orientée vers la connaissance pure, vers la contemplation). Sa qualité s'évalue en termes de *pertinence* (adéquation à l'action) et, si elle doit impérativement obéir à la logique, le respect formel de la logique ne suffit pas à garantir sa pertinence.

Pour pouvoir agir il ne suffit pas de percevoir : il faut encore *comprendre*. Le conducteur qui perçoit un feu rouge doit associer à cette perception une hypothèse causale : « ce feu rouge a été allumé pour indiquer qu'il fallait que je m'arrête ». Pour que nous puissions associer des hypothèses causales aux concepts, il faut que nous soyons équipés d'un *modèle* (ou *théorie*), c'est-à-dire d'un réseau de relations de causalité que nous postulons entre les concepts.

La compréhension n'est pas toujours instantanée : certains modèles comportant des paramètres, il faut les observer et cela prend du temps. Ainsi le médecin, avant de poser un diagnostic, observe sur son patient le poids, la tension, la température etc., puis raisonne pour faire un tri entre divers

6. Il ne convient pas de réserver le mot « expérience » au seul laboratoire où, pour poser des questions à la nature, on effectue des « expériences contrôlées » : le monde de la nature nous envoie, sans que nous lui posions de question, des événements qui se reflètent dans notre expérience et qui sollicitent notre action.

diagnostics possibles. La compréhension exige donc un *raisonnement* : ce raisonnement, intervenant dans l'utilisation d'un modèle, n'est cependant pas de même nature que celui qui permet de construire un modèle ; nous y reviendrons.

Parfois la compréhension est très rapide. Un judoka expérimenté place sa prise en une fraction de seconde, selon une action réflexe à laquelle il s'est préparé par des exercices répétés. De même, certaines personnes sont dotées du « coup d'œil » qui permet de faire fonctionner très vite, et avec une grande sûreté, le discernement et le raisonnement. Le « coup d'œil », comme le réflexe du judoka, se forme par l'entraînement et par une étude assidue ; il y faut aussi un talent qui n'est pas accordé à tous.

Ayant compris, nous pouvons *décider* : « on m'a signalé qu'il fallait que je m'arrête, se dit le conducteur, je décide de m'arrêter » puis *agir* pour produire, en réponse à l'événement du monde de la nature, un autre événement dans le monde de la nature : « j'appuie sur le frein » et, si la mécanique est fidèle, la voiture s'arrête (ne rien faire serait encore agir, mais par abstention⁷).

Telle est la boucle selon laquelle fonctionne notre relation avec le monde de la nature : événement - expérience - perception - compréhension - décision - action. Nous répondons à un événement en créant un autre événement.

* *

Cette boucle tourne vite. Pour la décrire nous avons dû l'immobiliser, tout comme on immobilise mentalement le moteur que l'on examine. Mais il faut en outre, pour concevoir son fonctionnement, se représenter sa dynamique. Elle comporte de nombreuses variantes.

Dans la boucle élémentaire que nous avons décrite, un événement du monde de la nature suscite, en réponse, une action qui constitue un autre événement dans le monde de la nature. Mais parfois l'action, fondée sur une anticipation de ses conséquences, provoquera le premier événement pour susciter une réponse du monde de la nature. Dans d'autres cas l'action suppose l'enchaînement de plusieurs opérations : elle progresse alors par étapes se concluant chacune par une action intermédiaire. Il arrive aussi que l'initiative de l'action appartienne à une personne, sa réalisation effective à une autre.

Se préparer à l'action demande une gestion de soi et un investissement⁸. Ainsi se construit l'*intelligence* qui est moins un talent inné qu'une maîtrise acquise de la charnière entre la pensée et l'action.

L'action est sujette à des pathologies. Les psychologues ont donné des noms évocateurs à certaines d'entre elles, que l'on rencontre dans la vie personnelle comme dans l'entreprise (Joule et Beauvais [9]) : « escalade d'engagement », « dépense gâchée », « piège abscons » etc. La boucle de l'action

7. « Le suicide est encore un acte » (Maurice Blondel (1861-1949), *L'action* (1893) p. VIII.)

8. La philosophie chinoise précise ces procédés de pensée : le sage est « disponible », « souple », « ouvert » ; il maîtrise les grilles conceptuelles, les modèles, mais ne se laisse emprisonner par aucun modèle (Jullien [95]).

peut se rompre dans chacune de ses étapes : une personne, ayant décidé, peut ne pas agir ; ayant compris, ne pas décider ; ayant perçu, ne pas comprendre ; ayant vu, ne pas percevoir.

Lorsque la volonté se dégrade en *velléité*, le passage de la décision à l'action est coupé. L'action est *inhibée* si c'est le passage de la compréhension à la décision qui est coupé. La longueur d'un enchaînement d'actions intermédiaires peut fait oublier son but final : alors l'action, dégradée en *activisme*, tourne à vide. Elle se dégrade enfin en *volontarisme* lorsque, rigidement liée à un modèle, elle veut ignorer l'expérience. La velléité, l'inhibition, l'activisme, le volontarisme sont dans nos entreprises des travers fréquents ; il importe de savoir les diagnostiquer et les rapporter à leur cause.

La grille conceptuelle équipe notre discernement mais aussi elle le limite ; elle nous emprisonne autant qu'elle nous outille. Nous ne pouvons pas discerner les événements qu'elle ne sait pas accueillir car ils se trouvent dans la « tache aveugle » de notre intellect (voir page 564). Lorsqu'un tel événement se produit, nous le voyons sous une forme trop vague pour pouvoir faire jouer le discernement, nous sommes comme le petit enfant qui ne voit que des taches colorées. Souvent nous préférons le juger sans importance et, s'il se manifeste avec insistance, nous nous détournerons de lui avec agacement. L'une des clés de la sagesse, c'est de considérer cet agacement comme un signal d'alarme : « si cet événement m'exaspère, il s'agit peut-être de quelque chose d'important », et on y regarde de plus près. Alors s'amorce le travail qui peut nous permettre de sortir de la prison mentale que constitue notre grille conceptuelle.

Une deuxième clé de la sagesse, c'est de gérer les associations d'idées qui se forment spontanément dans notre esprit. Elles sont, comme les mutations génétiques, souvent inefficaces ou nocives et en aucun cas elles ne peuvent servir d'étape à un raisonnement ; mais elles bousculent la grille conceptuelle et certaines d'entre elles sont fécondes : il faut savoir conserver celles-ci et éliminer les autres.

Nous ne disposons pas d'une seule grille, d'un seul modèle, mais de plusieurs correspondant chacun à une action différente : pour conduire notre voiture, pour faire la cuisine, pour la conversation, pour programmer etc. Nous passons d'un modèle à l'autre selon la situation et ce passage n'est pas immédiat : nous restons, pendant un délai, mentalement englués dans le modèle précédent. L'informaticien qui a passé une journée à programmer se trouve, le soir venu, incapable de lire le texte en langage naturel que lui propose un journal ; le mathématicien ou le joueur d'échecs, fascinés par un agencement compliqué d'abstractions simples, doivent se secouer pour percevoir les êtres humains qui les entourent et entendre ce qu'on leur dit. Une troisième clé de la sagesse consiste à associer, à la concentration, à l'enfermement dans un modèle, la vigilance qui maintient l'esprit disponible à ce qui survient dans le monde de la nature : de même la conduite automobile, activité routinière, doit comporter en « tache de fond » la vigilance qui permet de réagir en cas d'incident imprévu.

À l'intérieur de la boucle rapide de l'action tourne la boucle plus lente de la *réflexion* : c'est un investissement qui nous équipe en grilles conceptuelles

et en modèles, les uns et les autres adaptés à notre expérience. Si nous ne sommes pas bloqués par la fatigue ou par la conviction de détenir la vérité, cet investissement se poursuit pendant toute notre vie.

L'action répétée procure *l'expérience de l'expérience* : nous y reconnaissons des structures qui se répètent, des *patterns*. Cela nous incite à corriger nos modèles ou à en créer de nouveaux pour mieux comprendre et mieux agir.

La grille conceptuelle doit impérativement respecter des contraintes formelles (notamment le principe de non contradiction), puis sa qualité s'évaluera selon le critère de pertinence, d'adéquation à l'action. Pour la construire, il faut donc d'abord s'interroger sur le but que l'on vise, sur l'action que l'on entend accomplir. « Qu'est-ce que je veux *faire* ? » : c'est en partant de cette question que l'on peut définir des concepts pertinents et leur pertinence sera relative à cette action-là, à cette intention-là.

Ceux qui se lancent dans la construction d'un référentiel sans s'être posé cette question, et ambitionnent de décrire la réalité « telle qu'elle est », s'engagent dans une tâche sans issue parce qu'ils ne disposent d'aucun critère pour limiter la diversité des points de vue ni pour définir le degré de détail auquel il convient de s'arrêter.

La confrontation entre deux grilles conceptuelles portées par deux personnes différentes est souvent, sous une apparence logique et sémantique, la confrontation de deux intentions différentes. Elle ne peut être lucide que si l'on remonte aux *valeurs* qui orientent les intentions, si l'on élucide le « monde des valeurs » - mais il est rare que l'on y parvienne.

* *

Il est difficile de faire évoluer la grille conceptuelle car la grille ancienne est gravée dans les habitudes. La formation y aide en créant de nouvelles habitudes. L'émotion y contribue davantage encore : l'événement qui suscite une émotion se grave dans la mémoire et avec lui les concepts qui permettent de le discerner.

Pour faire évoluer la grille conceptuelle, il donc faut savoir gérer et ses émotions, et sa mémoire (Yates [221]; Squire et Kandel [97]; Carruthers [34]), de façon à sélectionner les éléments que l'on souhaite introduire dans celle-ci. Cela suppose de savoir méditer, ruminer les émotions. Certaines personnes en sont incapables : leur grille conceptuelle évoluera au hasard des événements, sous l'impulsion désordonnée des émotions.

Chacun s'équipe *volens nolens* d'une panoplie de modèles plus ou moins judicieux. Pour les construire il faut postuler des relations causales, puis si l'on est soigneux les tester (au moins par simulation mentale), les rejeter pour en choisir d'autres etc. jusqu'à ce que l'on soit parvenu à un modèle satisfaisant en regard de l'intention et de l'action (ce qui implique de respecter les exigences de la cohérence). Apprendre à lire, à écrire, à compter, à conduire une voiture, à faire la cuisine, à parler une langue, à programmer un ordinateur, c'est acquérir autant de modèles.

À cet apprentissage s'ajoute le dressage des réflexes, des habitudes, qui permettent de court-circuiter les étapes du raisonnement pour agir plus vite

Rôle de la philosophie

Quand on dit « mieux discerner », « mieux comprendre », « mieux agir », de quel « mieux » s'agit-il? Ou, pour parler comme l'économiste, quelle est la fonction objectif que l'on cherche à maximiser?

Pour l'expliciter il faut considérer les *valeurs* que l'on a l'intention de promouvoir. Or ces valeurs sont extérieures à l'action^a, tout comme en économie la fonction d'utilité du consommateur est extérieure à l'échange qu'elle conditionne.

La philosophie, quand elle s'applique à élucider les valeurs, éclaire leurs pathologies^b.

Les valeurs auxquelles chacun de nous adhère ne sont pas en effet strictement individuelles. Nous les héritons de notre éducation, de notre formation, de la culture dans laquelle nous sommes immergés. Or un tel héritage est trop composite pour être cohérent et, si nous poursuivons des valeurs incohérentes, notre vie sera désordonnée jusque dans sa racine.

L'incohérence des valeurs est révélée par ces « injonctions contradictoires » que la culture nous adresse et qui, si nous ne les dominons pas, mènent à la folie ou à la paralysie de l'action. Il faut que tu sois réaliste, mais les médias t'incitent à te complaire dans l'imaginaire; tu ne dois pas être violent, mais l'agressivité est un signe d'énergie; il faut que tu écoutes les autres, mais tu dois savoir t'affirmer; dans l'entreprise, on t'enjoint d'être à la fois discipliné et rebelle, soumis et créatif: comme si c'était possible!

S'il est vrai qu'il faut, selon les circonstances, savoir adopter des attitudes différentes - être en général paisible, mais violent si nécessaire - ce n'est pas à cette sage souplesse qu'invitent ces injonctions: elles nous confrontent à des règles qui, étant chacune absolue, sont incompatibles. Il ne s'agit pas de « contradictions dialectiques », d'aspects différents d'une même réalité qu'une synthèse pourrait assumer, mais d'*incohérences* qui détruisent la pensée et l'action parce que les valeurs elles-mêmes s'entre-détruisent.

Élucider les valeurs, c'est repérer les incohérences que comporte leur architecture, prendre conscience de la diversité de leurs origines, venir à leurs racines, les élaguer des considérations secondaires qui les encombrant, les reconstruire enfin pour conférer leur sens et à l'action, et à la pensée. *Cette élucidation, c'est la tâche primordiale de la philosophie.*

^a Les valeurs d'efficacité, d'élégance, de rapidité etc. sont elles aussi extérieures à l'action, même si elles s'y appliquent immédiatement.

^b S'il arrive à la philosophie d'aggraver la confusion des valeurs et donc de trahir sa mission, cela ne modifie en rien celle-ci.

et plus sûrement : l'intuition du mathématicien exercé lui fait entrevoir la démonstration au moment même où il lit l'énoncé d'un problème ; de même, l'expression de notre pensée en paroles, phénomène des plus complexes, se fait dans l'instant et le plus souvent sans grand effort.

* *

Il est déplorable que des mots comme « concept » ou « modèle » soient dans notre société réservés en fait à la démarche professionnelle du chercheur ou de l'intellectuel alors qu'ils désignent des choses que chacun fait tous les jours. En les prenant pour des choses abstraites, élevées, difficiles, on perd de vue le fonctionnement de notre propre pensée et on prend le risque de l'abandonner à des pathologies.

Les concepts, comme les modèles, n'ont pas d'autre but que d'équiper notre action, de servir notre volonté, d'incarner nos valeurs dans le monde ; ce sont des outils pratiques, orientés vers l'action, non des objets de pensée destinés à la contemplation. Il est vrai que la longueur de leur élaboration, ainsi que le délai qui sépare l'investissement de la mise en œuvre, font qu'ils peuvent sembler éloignés et comme détachés de l'action ; mais c'est là une illusion.

L'expérience nous confronte parfois à une situation imprévue face à laquelle il faut construire un modèle nouveau. Le monde ne se réduisant ni à un modèle, ni à la combinaison d'un nombre quelconque de modèles, de telles surprises sont inévitables. Ou bien l'on dispose d'un méta-modèle à partir duquel on pourra alors, par paramétrage et tri, construire rapidement le modèle adapté à cette situation ; ou bien il faut construire entièrement ce nouveau modèle et cela peut prendre des semaines, des mois, des années, délai pendant lequel l'action restera maladroite : il faut, pour construire entièrement un nouveau modèle, autant de temps que pour apprendre une langue étrangère.

La sagesse, avons-nous dit, réside dans l'aptitude à passer souplement d'un modèle à l'autre pour répondre aux exigences de la situation ; elle suppose aussi - et c'est la quatrième clé de la sagesse - d'investir dans la maîtrise du méta-modèle, du modèle des modèles, pour abréger le délai nécessaire à la mise au point d'un modèle nouveau.

* *

Pouvons-nous, pour comprendre l'entreprise, utiliser ce que nous avons appris sur la charnière dans la personne ? *A priori*, il semble que non : une entreprise n'est pas une personne, elle paraît plus complexe qu'une personne puisqu'elle organise un ensemble de personnes, de contrats, d'équipements, qui la situent à un carrefour de réseaux sociologiques, juridiques, physiques, économiques etc. Toutefois sa finalité est plus simple que celle d'une personne : produire efficacement des choses utiles, apporter le bien-être matériel au consommateur, c'est difficile mais moins que de tirer au clair les valeurs que l'on entend cultiver⁹...

9. L'entreprise a cependant elle aussi sa place dans le monde des valeurs : cela s'exprime

Ceux que le nomadisme de la carrière a conduit dans plusieurs entreprises savent que chacune a sa *personnalité* particulière, tout comme une ville ou un pays ont une personnalité que le voyageur peut percevoir. Pour désigner cette personnalité on utilise deux expressions : la « culture d'entreprise » désigne les valeurs professionnelles que partagent les personnels et la tonalité des rapports humains ; le « positionnement » désigne la place que l'entreprise occupe sur le marché, sa relation avec les segments de clientèle et les fournisseurs, sa situation par rapport à ses concurrents.

L'entreprise est ainsi dotée d'une individualité qui la circonscrit. Elle est en rapport avec un « monde extérieur » ou son action suscite des événements. Les artefacts qu'elle produit rejoignent le monde de la nature : une maison, un avion, une automobile, un programme informatique, ont appartenu d'abord en tant que projets au monde de la pensée, mais dès qu'ils sont produits ils deviennent pour l'action un point d'appui, à la fois obstacle et outil, et appartiennent dès lors au monde de la nature qui aura ainsi été transformé par l'action¹⁰.

L'entreprise élabore son action dans le « monde intérieur » de l'organisation et des processus de production. Tout comme l'action de l'individu, l'action de l'entreprise peut être réactive (réponse à un événement du monde extérieur) ou proactive (initiative qui crée l'événement en anticipant la réponse du monde extérieur). Elle suppose la mise en œuvre de concepts et de modèles. Les mots « gestion » et « investissement » s'y appliquent évidemment, désignant l'un les actions courantes, l'autre les actions différées.

Tout comme celle de la personne, l'action de l'entreprise peut comporter des délais : ainsi une entreprise industrielle alimente un stock en mobilisant les facteurs nécessaires à un programme de production qui résulte d'une anticipation de la demande, elle-même fondée sur l'analyse statistique du flux des commandes.

Mais alors que le langage, les concepts et les modèles qu'utilisent un individu résident dans sa tête, l'organisation de l'entreprise, son langage, ses concepts, ses modèles, ses processus résident aujourd'hui dans son système d'information, qui est désormais le lieu où se condensent le langage de l'entreprise ainsi que ses règles d'action.

Les concepts qui ne sont pas définis dans le système d'information se trouvent dans la « tache aveugle » de l'entreprise : les individus qui la composent peuvent les utiliser sans doute pour leur réflexion personnelle mais ils ne pourront pas les partager, les faire circuler. Si une personne s'avisait d'utiliser, pour son usage personnel, un référentiel des produits ou de l'organisation différent de celui qui réside dans le système d'information, les autres ne comprendraient pas ce qu'elle veut dire. Si cette personne ne réussit pas à faire modifier le référentiel de l'entreprise elle finira par renoncer à son référentiel personnel, même si d'aventure il était le plus pertinent.

dans le respect qu'elle accorde ou non à ses salariés, clients, fournisseurs et partenaires ; sous ce rapport, elle est aussi complexe qu'une personne.

10. Ce que l'on appelle « nature » en Europe lorsqu'on regarde un paysage est en fait le résultat d'une activité humaine (agriculture, élevage, jardinage) qui dure depuis des millénaires.

Le référentiel du système d'information délimite ce que l'on peut percevoir, penser et dire dans l'entreprise, tout comme la grille conceptuelle d'un individu délimite ce qu'il peut percevoir, penser et dire ; et il évolue aussi lentement, aussi péniblement, qu'une grille conceptuelle personnelle¹¹.

Toutes les entreprises voudraient aujourd'hui être souples et évolutives pour parer aux risques que comporte une économie instable et violente : elles ne pourront y parvenir que si elles s'appliquent à maîtriser leur référentiel, à organiser l'administration de leurs données, à maîtriser enfin leur système d'information.

* *

La réflexion sur l'action, les concepts et les processus se concrétise dans l'entreprise par la démarche dite d'urbanisation (voir page 433).

Urbanisation, modélisation des processus, référentiel et administration des données : telles sont les composantes qui dans l'entreprise sont l'équivalent de la réflexion, de la construction des modèles et de l'élaboration des concepts chez l'individu. Dans l'entreprise aussi la qualité des concepts est soumise au critère de pertinence ; la qualité des modèles, au critère d'efficacité ; concepts et modèles doivent servir l'action.

L'articulation de l'action et de la pensée peut, dans l'entreprise comme dans l'individu, être altérée par des pathologies (voir page 543) : il arrive que la complication du fonctionnement quotidien, ou les jeux de pouvoir, fassent perdre de vue les finalités de l'entreprise. Alors celle-ci se bureaucratise. C'est en examinant le système d'information que l'on peut porter sur l'entreprise le diagnostic le plus sûr, car ce système porte la trace des pathologies alors que le discours de l'organisation les cache.

6.2 Complexité et complication

« Ce qui est simple est toujours faux. Ce qui ne l'est pas est inutilisable. » Paul Valéry (1871-1945), *Mauvaises pensées et autres*, 1942, in *Œuvres*, Tome II, Gallimard, Bibliothèque de La Pléiade 1960, p. 864.

« L'essence des mathématiques (...) apparaît comme l'étude des relations entre des objets qui ne sont plus (volontairement) connus et décrits que par quelques-unes de leurs propriétés, celles précisément que l'on met comme axiomes à la base de leur théorie. » (N. Bourbaki, [21], p. 33).

Quand on prononce le mot « complexité », un ensemble confus de notions affleure : théorie du chaos, théorème de Gödel, principe d'incertitude de Heisenberg, fractales, limites du calcul informatique etc. Chacune est

11. L'évolution est lente mais comme elle se fait par à-coups elle peut comporter des épisodes de changement rapide, tout comme la tectonique des plaques peut provoquer des séismes.

claire dans son domaine propre, c'est l'accumulation qui crée la confusion (Bouveresse [23]). Tentons de donner au mot « complexité » une acception qui la dissipera (Volle [214]).

Le monde de la nature (y compris de la nature humaine et sociale) qui se présente à la perception et à la pensée est *concret* en ce sens qu'il se présente *hic et nunc*, son individualité se manifestant dans des particularités de temps et de lieu.

Aucune pensée ne peut rendre compte de l'ensemble des propriétés du monde de la nature. Il suffit pour s'en convaincre de considérer une tasse de café et de tenter de la décrire. Chacune de ses propriétés relève d'un schéma conceptuel (donc abstrait) : sa forme géométrique, à la précision de laquelle on ne peut assigner aucune limite ; ses origines culturelles, économiques, industrielles ; sa composition chimique ; la position et les mouvements des molécules, atomes, particules qui la composent¹² etc. Chaque objet concret assure ainsi *de facto* la synthèse d'un nombre illimité de déterminations abstraites. Il est en toute rigueur *impensable*, et c'est cela que transcrit l'adjectif « complexe ». Il en est de même du monde lui-même, ensemble des objets concrets.

Sur chaque objet concret, nous disposons non d'une connaissance complète mais de « vues » dont chacune permet de considérer l'objet à travers une grille conceptuelle particulière. Si je ne peux parler d'une mesure précise de ma tasse de café, toute mesure étant grossière par rapport à un ordre de précision supérieur, je peux dire que la mesure est « exacte » si elle me permet de faire sur l'objet un raisonnement exact, c'est-à-dire adéquat à l'action que j'ai l'intention de réaliser : je peux calculer l'ordre de grandeur de sa densité à partir de mesures approximatives de sa masse et de son volume, inférer de l'examen de sa composition chimique une évaluation qualitative de sa fragilité... ou simplement boire mon café.

L'objet étant sujet à un nombre illimité de déterminations, il existe un nombre illimité de « vues » toutes logiquement équivalentes. Cependant certaines seront plus utiles en pratique pour un sujet placé dans une situation particulière, que ce sujet soit individuel ou social : ce sont les vues en relation avec l'action du sujet, avec l'articulation entre sa volonté et l'objet considéré comme obstacle ou comme outil. Ces vues-là sont « pertinentes » ainsi que les observations et raisonnements que le sujet peut faire en utilisant les catégories selon lesquelles elles découpent l'objet.

Ainsi le spectacle d'une rue conjugue des déterminations historiques, architecturales, sociologiques, économiques, urbanistiques, physiques, esthétiques etc. Cependant le conducteur d'une automobile limite son observation à quelques éléments : signalisation, bordures de la voie, obstacles dont il estime la vitesse et anticipe les déplacements. Cette grille fait abstraction de la plupart des aspects de la rue mais elle est adéquate à l'action « conduire l'automobile » : le conducteur qui prétendrait avoir de la rue une représentation exhaustive saturerait sa perception et serait un danger public.

12. Si l'on recherche une précision de l'ordre de l'Angström (10^{-10} m), la connaissance simultanée des positions et vitesses est bornée par le principe d'incertitude de Heisenberg.

Nous trouvons « naturelles » nos grilles habituelles, nous qualifions d'« objectives » les observations réalisées selon ces grilles. Pourtant la façon dont la pensée découpe ses concepts évolue selon les besoins et elle est, en ce sens, subjective :

1) La classification des métiers et niveaux de formation, « concrète » pour les personnes dont elle balise la carrière, est une grille conceptuelle et donc en fait une abstraction (Desrosières et Thévenot [202]) : la catégorie des « cadres », qui appartient au langage courant en France, n'existait pas avant les classifications Parodi de 1945.

2) La classification des êtres vivants a évolué de Linné, Jussieu et Darwin à la « cladistique » contemporaine (Lecointre et Le Guyader [78]). Fondée sur la comparaison génétique, cette dernière a introduit des bouleversements : le crocodile est plus proche des oiseaux que des lézards ; les dinosaures sont toujours parmi nous ; les termes « poisson », « reptile » ou « invertébré » ne sont pas scientifiques.

3) Les classifications de l'industrie (Guibert, Laganier et Volle [77]) ont pris pour critère au XVIII^e siècle l'origine de la matière première (minérale, végétale, animale) conformément à la théorie des physiocrates et aux besoins d'une économie dont la richesse provenait, pour une large part, de l'agriculture et des mines. Au milieu du XIX^e siècle les controverses sur le libre échange ont conduit à un découpage selon le produit fabriqué. À la fin du XIX^e siècle le critère dominant fut celui des équipements : le souci principal était alors l'investissement. Depuis la dernière guerre les nomenclatures sont construites de façon à découper le moins possible les entreprises (« critère d'association ») car l'attention s'est concentrée sur les questions d'organisation et de financement.

4) Au XVI^e siècle il paraissait normal de regrouper les faits selon des liens symboliques : pour décrire un animal le naturaliste évoquait son anatomie, la manière de le capturer, son utilisation allégorique, son mode de génération, son habitat, sa nourriture et la meilleure façon de le mettre en sauce (Foucault [61] p. 141). Plus près de nous, il a fallu du temps pour réunir les phénomènes magnétiques et électriques, puis reconnaître la nature électromagnétique de la lumière, enfin pour réunir masse, énergie et lumière.

6) Dans l'entreprise, les classifications des produits, clients, fournisseurs et partenaires, ainsi que la définition des rubriques comptables, évoluent avec les besoins. C'est pourquoi le référentiel de l'entreprise est *centrifuge* : s'il n'est pas contrôlé, il se dégrade en variantes et les données deviennent incohérentes. L'insouciance de la plupart des entreprises en matière d'administration des données résulte d'une erreur de jugement : comme on croit les classifications « naturelles », on ne voit pas à quel point elles sont instables et on sous-estime l'entropie qui mine la qualité du système d'information (voir page 79).

Les grilles à travers lesquels nous percevons le monde nous en donnent une vue sélective ; il s'agit d'un *langage* qui évolue plus ou moins vite selon les domaines (les classifications de la science ou de la vie courante changent moins souvent que celles de l'entreprise). Ainsi le cadre conceptuel que nous

utilisons est *construit* ; il porte la trace de choix pour partie intentionnels, pour partie conventionnels.

Cela ne veut pas dire que les *faits* eux-mêmes soient construits, comme le pensent trop vite les apprentis philosophes. En effet si tout cadre conceptuel, même pertinent, ne reflète le monde que de façon partielle, ce reflet n'en est pas moins *authentique*. L'automobiliste qui arrive à un feu de signalisation ignore les détails de l'architecture des immeubles alentour mais il voit ce feu, ce qui lui permet de l'interpréter et d'agir. Même si sa grille ne révèle pas la Vérité du Monde elle lui permet de savoir si le feu est vert, orange ou rouge. La couleur du feu ne relève plus alors d'une hypothèse mais constitue un *fait d'observation* dont il peut et doit tirer les conséquences pratiques.

Si l'observation ne peut pas être exhaustive, elle peut être *exacte*, c'est-à-dire suffire pour alimenter un raisonnement exact. Celui-ci peut très souvent se satisfaire d'ordres de grandeur, ce qui détend l'exigence de précision. La réalité, si elle n'est pas pensable dans l'Absolu, est ainsi en pratique pensable pour l'action, pour vivre dans le monde et y graver nos valeurs comme nos ancêtres ont gravé les symboles de leurs mythes sur les parois des grottes.

* *

« (Nature's) fundamental laws do not govern the world as it appears in our mental picture in any direct way, but instead they control a substratum of which we cannot form a mental picture without introducing irrelevancies. » (Paul Dirac, *The Principles of Quantum Mechanics*, introduction, Oxford, Clarendon Press, 1930)

« Le théorème de Gödel (...) est certainement de beaucoup le résultat scientifique qui a fait écrire le plus grand nombre de sottises et d'extravagances philosophiques. » (Jacques Bouveresse, [23] p. 60)

Le monde des modèles, le monde de la théorie, c'est le *monde de la pensée*, qui est aussi celui de nos artifices, jeux, langages de programmation et programmes informatiques, de nos machines (en tant qu'objets concrets elles appartiennent au monde de la nature, mais leur conception relève du monde de la pensée) et de nos organisations (même remarque).

Le monde de la pensée, des concepts et propositions que l'on peut échauffer en obéissant au principe de non contradiction, est complexe : il est impossible d'en rendre compte à partir d'un nombre fini d'axiomes. Cependant toute pensée explicite, même subtile, est *simple* - non dans son processus d'élaboration, qui étant concret est complexe, mais dans le *résultat* de ce processus.

Alors que tout objet concret relève d'un nombre illimité de déterminations, toute pensée explicite s'exprime en effet selon un nombre fini de concepts. Toute pensée visant à l'action met en œuvre un *modèle* (ou *théorie*) constitué par le couple que forment d'une part un découpage conceptuel de l'observation, d'autre part des *hypothèses* sur les relations fonctionnelles.

Que le modèle soit formalisé, explicite, pertinent ou non, cette démarche est générale. Toute observation est une mesure selon une grille définie *a priori* ; tout raisonnement suppose que l'on prolonge cette mesure en postulant des relations fonctionnelles entre les concepts (Korzybski [106]).

* *

La pensée a un but lointain : se confronter avec le réel dans l'*expérience* lors de laquelle les concepts seront soumis au critère de pertinence, les théories à l'épreuve de la réfutation. Mais il existe un moment où la pensée se forme sans être confrontée à l'expérience, se muscle comme le font en jouant les jeunes animaux.

La pensée pure dispose pour se préparer à l'expérience d'une arme puissante : le *principe de non contradiction*. Toute théorie comportant une contradiction est fautive en ce sens qu'il ne pourra pas exister d'expérience à laquelle elle s'appliquerait. Le monde de la nature étant non contradictoire, *le viol de la logique est contre nature* : une chose ne peut pas à la fois être et ne pas être, posséder une propriété et ne pas la posséder. Cela n'exclut pas qu'elle puisse évoluer ou encore posséder des facettes différentes comme une feuille de papier qui serait blanche d'un côté, noire de l'autre : les paradoxes résultent des imprécisions du langage courant.

Le fonctionnement de la pensée pure est un jeu avec des hypothèses. Pour pratiquer cette gymnastique, il faut poser des hypothèses et explorer leurs conséquences, puis recommencer etc. Celui qui ne s'est pas préparé ainsi posera des hypothèses naïves et s'aventurera dans des impasses théoriques que les experts ont appris à éviter. Le but des mathématiques n'est autre que cette gymnastique de l'esprit.

La non contradiction est une *garantie de réalisme potentiel*. Les géométries non euclidiennes, construites de façon formelle et sans souci d'application, ont par la suite fourni les modèles pour représenter des phénomènes physiques. Toute théorie non contradictoire peut ainsi trouver dans la complexité du monde de la nature un domaine d'application (mais le caractère non contradictoire d'une théorie ne garantit pas sa pertinence face à une situation particulière)¹³. La pensée pure n'est donc pas seulement une gymnastique : c'est un investissement qui procure des modèles en vue des expériences futures.

La conquête de la pensée pure, c'est l'*intelligence formelle*, maîtrise du raisonnement qui, partant de données initiales, va droit au résultat. Lorsque l'esprit a parcouru plusieurs fois un raisonnement il l'anticipe comme l'on anticipe les formes et le contenu d'un appartement familial ; il l'enjambe pour en construire d'autres plus généraux, plus abstraits. La portée du raisonnement s'élargit alors comme un cercle lumineux. Des champs entiers de la pensée s'articulent à un principe qui a été conquis par un héroïque effort

13. Si l'on suppose que la complexité du monde de la nature ne connaît pas de limite, alors il est certain qu'il existe, pour chaque théorie non contradictoire, au moins un phénomène naturel qu'elle aiderait à modéliser. Si ce n'était pas le cas, en effet, cela voudrait dire que la complexité du monde de la nature est limitée par l'exclusion de tout phénomène modélisable selon cette théorie.

d'abstraction : principe de moindre action en physique (Landau et Lifchitz [117] p. 8) ; optimum de Pareto en économie (Ekeland [52] p. 59) ; « voile d'ignorance » en éthique (Rawls [170]) ; principe de non contradiction en logique et en mathématique (Bourbaki [20]).

L'intelligence formelle, dont le terrain propre est la pensée pure, s'exerce pendant la jeunesse. Certains adolescents sont des mathématiciens de génie comme Galois ou de grands joueurs d'échecs.

* *

Mir hilft der Geist ! auf einmal seh' ich Rat
Und schreibe getrost: Im Anfang war die *Tat* ¹⁴!
(Goethe, *Faust*, 1808, vers 1236-1237)

Le jeu de la pensée pure reste puéril s'il n'aboutit pas à la confrontation au monde dans l'action. L'esprit formé au jeu avec des hypothèses trouve ici du nouveau à apprendre : face à la situation particulière à laquelle le sujet est confronté *hic et nunc*, et compte tenu de sa volonté (vivre et cultiver ses valeurs), que doit-il *faire* ? Ne pas agir serait encore une action, fût-ce par abstention (Blondel [16]). Pour choisir l'action à engager il faut que le sujet puisse anticiper ses conséquences, donc dispose d'un modèle du monde de la nature dans lequel il fera par la pensée simuler l'effet de son action.

Il doit alors, dans la batterie des hypothèses avec lesquelles il jouait librement, choisir celles qui représenteront le monde de la nature avec exactitude en regard des impératifs de son action. L'expérience oblige ainsi à renoncer à certaines hypothèses et à en retenir d'autres ; elle tourne le dos à la liberté qui caractérisait la pensée pure. Les êtres humains ont longtemps pu se représenter la surface de la Terre comme un plan infini, un disque ou une sphère, hypothèses alors également plausibles ; puis la pratique de la navigation et l'expérience de l'astronomie ont imposé la troisième hypothèse. C'est un moment émouvant que celui où l'esprit se courbe sous le joug de l'expérience.

L'expérience prouve-t-elle la vérité des hypothèses ? Oui, s'il s'agit de *faits* que tranche l'observation, comme la sphéricité approximative de la Terre, la mesure de la distance moyenne entre la Terre et le Soleil, la date d'un événement. Non, s'il s'agit de relations fonctionnelles entre concepts : lorsque nous postulons la vérité d'une hypothèse causale que l'expérience a validée, nous inférons une proposition générale à partir d'une expérience limitée et cette induction n'est pas une preuve.

Il en résulte, selon Popper [164], que toute théorie doit être présentée de telle sorte que l'on puisse la réfuter par l'expérience : le scientifique doit être assez modeste pour préparer dans ses travaux les voies de leur réfutation. Toute théorie construite de façon qu'on ne puisse pas la réfuter est nulle *en raison même de sa solidité apparente* (les faits d'observation sont, eux, irréfutables mais ils ne constituent pas des théories). Lorsque l'expérience

14. « L'Esprit m'aide ! Soudain j'y vois clair et j'écris tranquillement : au commencement était l'*action* ! »

réfute la théorie, elle le fait d'une façon toujours logique mais surprenante. Ces surprises sont son apport le plus précieux¹⁵.

* *

Si la gymnastique de la pensée est analogue aux jeux des jeunes animaux, la pratique de l'action est analogue à la recherche des ressources (chasse, pâturage) et des partenaires sexuels par les animaux adultes, recherche à laquelle l'être humain ajoute le besoin d'exprimer ses valeurs par des symboles. *La démarche expérimentale caractérise l'âge adulte de la pensée*. L'essentiel pour l'adulte n'est pas en effet l'intelligence formelle, même si elle lui est nécessaire, mais l'efficacité dans l'action. Il y applique son discernement (découpage des concepts pour distinguer les êtres observés) et son jugement (sélection du modèle pertinent). Il y engage spontanément la capacité intellectuelle acquise lors des jeux de l'enfance.

L'expérience de l'expérience, la confrontation répétée à des situations nécessitant des modèles divers, assouplit et accélère la construction théorique. Au sommet de l'art, l'adulte acquiert le « coup d'œil » : face à la complexité et l'urgence d'une situation particulière il va droit à l'action nécessaire. L'esprit enjambe alors les étapes d'un raisonnement qu'il ne se soucie pas d'explicitier.

Si le sage chinois est « sans idée » (Jullien [95]), ce n'est pas parce qu'il a l'esprit vide ou qu'il ne s'intéresse pas à l'action : disposant de modèles divers, il passe de l'un à l'autre pour s'adapter à la situation, obéir à la « propension des choses » et atteindre un sommet d'efficacité (Jullien [93]). S'il ne s'attache à aucun modèle, c'est qu'il sait à chaque moment mobiliser celui qui convient, voire en conjuguer plusieurs. Cet idéal de sagesse, impossible à réaliser complètement, brille à l'horizon comme un point lumineux et indique le chemin de l'ambition pratique la plus haute, le *Tao* : être disponible devant le monde afin d'y être efficace par l'action.

On évoque souvent le « coup d'œil » du stratège militaire, soumis à des contraintes extrêmes. On le rencontre aussi chez les entrepreneurs, artisans, contrôleurs aériens, pilotes d'avion, conducteurs automobiles, sportifs, chirurgiens, bref chez tous ceux qui doivent agir.

* *

Certains des obstacles qui s'opposent à la pensée adulte, à la pensée appliquée à l'action, sont naturels : il est naturel qu'un débutant soit maladroit. D'autres obstacles, par contre, constituent un handicap qui empêche de se former par l'exercice et finalement interdit l'action. Mais alors que l'animal prédateur qui ne sait pas chasser meurt bientôt, nos sociétés élaborées pro-

15. Feynman a illustré ainsi les surprises que l'on rencontre en physique des particules : sur un échiquier, les blancs ont deux fous dont l'un joue sur les cases noires, l'autre sur les cases blanches. Il est raisonnable d'anticiper que durant la partie ces fous joueront sur des couleurs différentes. Supposons cependant que le fou qui joue sur les cases blanches se fasse prendre, puis qu'un pion blanc aille à dame sur une case noire et que le joueur lui substitue ce fou : alors les blancs auront deux fous sur les cases noires. Cette situation résulte d'un concours de circonstances rare mais non impossible, et qu'il serait difficile d'imaginer *a priori*.

duisent en nombre des personnes qui ne savent pas agir ou seulement dans des domaines limités. Certaines, bien qu'intelligentes sont incapables d'agir ; d'autres, comme dotées d'une sagesse à éclipses, sont aptes à l'action dans leur vie personnelle mais non dans leur vie professionnelle ou inversement.

Il se peut que cette mutilation contribue à la reproduction de la société tout comme la stérilité des ouvrières contribue à la reproduction de la ruche. Le constat d'une mutilation si fréquente étant douloureux, celui qui énonce ce grand secret est mal reçu. Tâchons d'en élucider le mécanisme.

* *

L'écart entre la pensée et le monde n'a rien de scandaleux ni de bouleversant. Nous sommes incapables de décrire le mécanisme neurophysiologique qui nous permet de prononcer la lettre « A » (Leibowitz [111]) ou de décrire un visage par des paroles ; le fonctionnement quotidien de notre corps reste énigmatique ; si nous nous intéressons passionnément à la personne aimée, sa connaissance n'est jamais achevée : étant concrète, cette personne est aussi complexe que le monde lui-même.

Si l'écart entre la pensée et le monde fait pourtant souffrir, cela vient d'une formation intellectuelle mal conçue : les adultes font croire à l'adolescent que le monde de la pensée est aussi éloigné de la vie courante que peut l'être le paradis ; devenu adulte, il ne concevra pas comment la pensée peut devenir un outil pratique et servir de levier à l'action dans un monde complexe.

On peut se demander si certaines pédagogies n'ont pas pour effet (et, de façon perverse, pour but) de stériliser les esprits en leur inculquant devant les choses de l'intellect une humilité déplacée : s'il faut être modeste devant le monde que l'on découvre par l'expérience, chacun a en effet le devoir d'être intrépide dans la pensée¹⁶.

* *

Les personnes mal formées croient que la tâche de la pensée est de représenter le monde tel qu'il est. Toute pensée exprimée avec simplicité leur semble alors une usurpation : la simplicité montrant naïvement que cette pensée est incapable de représenter le monde, elles estiment que celle-ci ne vaut rien et n'a donc pas même le droit d'être exprimée. À la pensée qui laisse apparaître sa simplicité elles préféreront une pensée *compliquée*.

La pensée compliquée est simple en fait comme toute pensée en ce sens qu'elle repose sur un nombre fini de postulats, mais elle prend soin de cacher sa simplicité derrière un écheveau de concepts et relations fonctionnelles dont l'architecture embrouille postulats, conséquences, résultats intermédiaires et hypothèses annexes.

16. « L'une des raisons principales qui éloignent autant ceux qui entrent dans ces connaissances du véritable chemin qu'ils doivent suivre, est l'imagination qu'on prend d'abord que les bonnes choses sont inaccessibles, en leur donnant le nom de grandes, hautes, élevées, sublimes. Cela perd tout. Je voudrais les nommer basses, communes, familières : ces noms-là leur conviennent mieux ; je hais ces mots d'enflure... » (Pascal, [151], in *Œuvres complètes*, Gallimard, Bibliothèque de La Pléiade 1954 p. 602)

La pensée compliquée est, en pratique, inutilisable. Il arrive que sous sa complication se cache une incohérence : alors la pensée est non seulement inutilisable mais elle s'annule d'elle-même.

Les contraintes formelles de la rédaction des textes scientifiques, excellentes en elles-mêmes, permettent à des esprits faibles de publier des écrits dont le vide est masqué par la complication : c'est ce que Feynman appelait *pretentious science*¹⁷.

La complication du modèle singe la complexité du réel. Elle n'égale jamais la complexité du réel mais elle sature l'attention et le jugement. La personne qui examine un modèle compliqué est en « surcharge mentale » : le modèle lui semble alors aussi complexe qu'un objet réel.

Un modèle compliqué sera cependant considéré avec respect par les personnes qui se défient de la simplicité et ne jugent pas nécessaire de comprendre ce qu'elles lisent. Elles le croient *réaliste*, et en effet une des façons de construire un modèle compliqué, c'est d'emprunter à la réalité un grand nombre de déterminations à partir desquelles on emmêlera un écheveau.

Un modèle simple est par contre vulnérable dans toutes ses étapes puisqu'elles sont compréhensibles ; il est scientifique au sens de Popper. Mais celui qui présente un modèle simple s'attire souvent la phrase « ce n'est pas si simple ! ».

Considérons le cas du système d'information. 80 % des fonctionnalités développées à grands frais, et dont la maintenance sera elle aussi coûteuse, ne sont pas utilisées. Comment expliquer ce gâchis ?

Tout système d'information est fondé sur une *abstraction* : il représente les êtres qu'il considère (clients, fournisseurs, produits, agents, entités de l'organisation etc.) par des classes dotées d'un nombre fini d'attributs et de règles de gestion ; les valeurs des attributs sont codées selon des nomenclatures choisies en fonction des besoins. La spécification des attributs et des règles élimine, par son silence, les attributs qui ne seront pas observés, les règles qui ne seront pas appliquées. Cette simplification est intolérable pour les personnes qui aiment la pensée compliquée. Elles iront chercher les cas particuliers qui ne se coulent pas dans le modèle, et exiger qu'on le complique sous prétexte que l'informatique doit se plier à la demande des utilisateurs (idée juste utilisée ici de façon perverse).

Ces personnes trouveront trop simple de coder une classification selon une suite de partitions emboîtées. Elles vont préférer que les rubriques d'un même niveau se chevauchent, que les niveaux ne s'emboîtent pas exactement. Très sensibles à la solidarité entre les diverses parties du monde, elles pensent

17. « The work is always: (1) completely un-understandable, (2) vague and indefinite, (3) something correct that is obvious and self-evident, worked out by a long and difficult analysis, and presented as an important discovery, or (4) a claim based on the stupidity of the author that some obvious and correct fact, accepted and checked for years is, in fact, false (these are the worst: no argument will convince the idiot), (5) an attempt to do something, probably impossible, but certainly of no utility, which, it is finally revealed at the end, fails or (6) just plain wrong. There is a great deal of "activity in the field" these days, but this "activity" is mainly in showing that the previous "activity" of somebody else resulted in an error or in nothing useful or in something promising. » (Gleick [70] p. 353)

que tout est relié à tout : elles diraient volontiers que le fonctionnement du système solaire est sensible à l'attraction des étoiles. Elles s'opposent donc à la modularité du système d'information et militent pour qu'il traite en bloc les divers aspects du métier, ce qui accroît la taille des projets et complique leur réalisation.

Dans un système d'information, la logique voudrait que les tables de codage fussent identiques pour toutes les applications. La sociologie de l'entreprise, le particularisme des métiers, l'insouciance des dirigeants, les circonstances de l'exécution font cependant qu'en pratique l'architecture des bases de données n'est jamais cohérente. Elle est soumise à un phénomène d'entropie irrésistible dont l'explication réside dans la nature même des données (Boydens [25]) :

1) L'interprétation des données en informatique scientifique (chimie, biologie etc.) évolue et comporte des ambiguïtés sémantiques, même si ces données sont vérifiées et contrôlées.

2) Il est normal qu'un agent opérationnel fasse passer les exigences de son travail avant les tâches de saisie¹⁸ ; mais il en résulte qu'en informatique de gestion les données sont souvent incomplètes. En outre parmi les données saisies seules celles que l'agent juge importantes auront été bien vérifiées¹⁹. Il arrive aussi que des interprétations locales soient données aux tables de codage, qui se dégradent alors en dialectes.

3) Il y a conflit entre l'exigence formelle du code informatique et le flou inhérent à des concepts dont l'interprétation est sujette à l'expérience humaine, même quand il s'agit de concepts générateurs de droits et de devoirs (cotisations, prestations sociales, impôts etc.) : la distinction entre un ouvrier et un employé repose sur le caractère prépondérant de leurs activités manuelles ou intellectuelles, qu'il est bien difficile d'évaluer ; des difficultés analogues se rencontrent avec les concepts de journée de travail, de catégorie d'activité etc.

Les codages se diversifient dans le temps et l'espace, tout raisonnement doit passer par une phase pénible de retraitement des données. Les statistiques issues de sources différentes sont incohérentes, car elles mesurent des réalités différentes. Les tableaux de bord occasionnent de pénibles discussions en comité de direction : « D'après mes données ça monte, et vous dites que ça baisse ? À la réflexion, cela doit provenir du fait que j'ai consolidé telle filiale alors que vous vous référez à un autre périmètre, etc. »

* *

La réalité n'est jamais aussi simple qu'un modèle, quelle que soit la richesse de celui-ci, puisqu'elle est complexe alors que le modèle est fini. La protestation « ce n'est pas si simple ! » est donc vide de sens car elle s'applique à tout modèle, fût-il compliqué.

18. Le conseiller de l'ANPE qui vient de trouver un emploi pour un chômeur serait mal venu de retenir celui-ci par la manche pour « finir de remplir le dossier » : le dossier reste incomplet pour une raison parfaitement admissible.

19. Il est naturel que le contrôleur qui vérifie une déclaration fiscale examine soigneusement les données qui déterminent l'assiette de l'impôt et soit moins attentif aux autres.

La simplicité de la pensée est un outil pour l'action comme l'imperfection de la mémoire est un outil pour l'intellect. L'oubli sélectif suscite le travail de synthèse et exerce l'intelligence : tout garder en mémoire, c'est ne rien comprendre (Squire et Kandel [97]). De même tout percevoir, c'est ne rien pouvoir faire.

La question que l'on doit se poser n'est donc pas « ce modèle est-il réaliste » puisqu'il ne peut pas l'être, mais « ce modèle fournit-il une simplification pertinente », qui permette de raisonner juste et d'agir efficacement (Fixari [58]). Ceux qui refusent la simplicité du modèle refusent l'apport le plus précieux de la pensée : la sélection qu'elle opère dans la multiplicité illimitée des phénomènes pour n'en retenir que la vue pertinente, qui permet l'action efficace.

Pour sortir de l'embarras, il faut assumer et cultiver la simplicité de la pensée. Nous aurons fait un grand progrès lorsque nous rirons de celui qui dit « ce n'est pas si simple ! », « il faut bien qu'on réponde à la demande des utilisateurs » ou tout autre phrase qui révèle le refus de la simplicité de la pensée. On peut aussi s'appuyer sur quelques outils et *procédés de pensée* : modèle en couches, croisement des découpages, raisonnement probabiliste etc.

Modèle en couches

Un modèle en couches (voir page 42) consiste en l'articulation de plusieurs sous-modèles caractérisés chacun par un protocole spécifique et reliés par des interfaces. Il permet de représenter les situations où plusieurs logiques jouent simultanément. Le nombre des logiques ainsi articulées restant fini, le modèle n'atteint pas la complexité du monde de la nature, mais tout en restant pensable il possède un des traits de cette complexité : la pluralité des logiques.

Son domaine d'application est vaste. Il peut servir pour décrire les télécommunications, le transport aérien, modéliser les systèmes d'information, le fonctionnement de l'ordinateur etc.

Croiser les découpages

On peut considérer un même objet selon diverses grilles. « Croiser les découpages », c'est considérer un tableau croisé (ou un « hypercube ») qui ventile l'objet selon deux ou plusieurs grilles et permet d'examiner leurs corrélations (voir page 207).

Imprévisibilité et probabilité

Au début du XIX^e siècle les sciences physiques supposaient que l'évolution d'un système était déterminée une fois connues les positions et vitesses initiales ; l'avenir était prévisible. Ce déterminisme a étendu son empire sur des domaines comme l'histoire, l'économie, la sociologie où sa pertinence est cependant douteuse. Le choc a été profond lorsque la physique elle-même lui a imposé des limites : à l'échelle subatomique, le modèle de la mécanique quantique est probabiliste.

On aurait cependant pu s'aviser qu'à l'échelle de la vie courante le déterminisme est contredit par l'expérience la plus banale : si les physiciens pouvaient prédire le résultat d'un coup de dés, cela se saurait dans les salles de jeu ; si on pose un crayon bien aiguisé en équilibre sur sa pointe, il tom-

bera bientôt en raison des chocs qu'il reçoit de la part des molécules de l'air environnant mais il est impossible de prévoir l'angle de sa chute.

Les phénomènes régis par des équations différentielles non linéaires, bien que déterministes par nature, donnent naissance à des effets chaotiques que l'on ne peut pas distinguer d'un comportement probabiliste car ils sont, comme le lancement d'un dé, très sensibles aux conditions initiales (Gleick [69]). Ainsi on ne peut pas garantir que la Terre ne quittera jamais le système solaire : la prévision de sa trajectoire comporte une incertitude qui croît à mesure que l'on s'éloigne dans le futur.

Pour traiter l'incertitude, la science économique a créé la théorie des anticipations et du risque. Un entrepreneur raisonne en avenir incertain. Il en est de même du stratège qui doit décider juste alors qu'il reçoit des rapports partiels, erronés ou fallacieux. Il existe des généraux qui gagnent les batailles et des dirigeants efficaces : ce sont ceux qui, possédant le « coup d'œil », savent agir au mieux dans les situations incertaines²⁰. Cette faculté s'acquiert par l'entraînement et ceux qui la possèdent n'ont le plus souvent ni le goût, ni la possibilité d'expliquer leurs raisonnements : l'action juste s'impose à eux comme une nécessité évidente.

Limites de la logique

Au début du xx^e siècle Bertrand Russell et Alfred Whitehead se sont efforcés de donner aux mathématiques un fondement à la fois logiquement correct et complet. Kurt Gödel [63] a démontré en 1931 que quel que soit le système d'axiomes que l'on retient pour fonder une théorie, il existe des propositions que l'on sait vraies mais dont la vérité ne peut pas être démontrée dans le cadre de cette théorie (voir page 85). Ainsi aucun système d'axiomes ne peut égaler la complexité du contenu potentiel de la pensée, et la logique ne peut pas avoir réponse à tout. Certains logiciens s'opposaient à cette affirmation avec une certaine raideur. Avec Gödel, la logique a rencontré sa propre limite en s'appuyant sur ses propres méthodes.

La pensée potentielle, constituée des propositions que l'on pourrait déduire de l'ensemble des systèmes axiomatiques possibles, est donc complexe ; mais la pensée explicite, résultat de nos réflexions, est fondée sur un nombre fini d'axiomes : Gödel a démontré que la pensée explicite était plus simple que la pensée potentielle.

La cohérence est condition *nécessaire* de l'efficacité pratique de la pensée, car une pensée incohérente est pratiquement infirme ; mais le caractère logique d'un système ne prouve pas sa pertinence face à une situation particulière : un délire peut être cohérent.

Écoute

Comment élaborer l'adéquation à l'action ? Pour comprendre une situation particulière, faire le choix des concepts pertinents, élaborer une théorie exacte, la pensée pure ne suffit pas. Lorsqu'on veut construire un système d'information, l'*écoute* est non seulement convenable au plan moral, mais pertinente en tant que démarche.

20. Napoléon disait « j'aime les généraux qui ont de la chance ».

Durant la phase d'écoute, la grille conceptuelle de l'auditeur est mise entre parenthèses (sauf la partie de la grille qui est propre à l'écoute elle-même) ; il accepte de s'engager dans un voyage mental vers des constructions intellectuelles qui ne lui sont pas familières. Il y faut de la modestie : celui qui entre dans un domaine nouveau est un bizut et se fait bousculer par les experts.

Après l'écoute vient la synthèse : il ne faut pas croire tout ce que l'on entend ; les habitudes des praticiens sont parfois illogiques car elles découlent de la superposition de méthodes anciennes souvent dogmatisées. L'auditeur rend les incohérences visibles. Quand le « bizut » se familiarise ainsi, et commence à parler avec quelque autorité, les spécialistes n'apprécient guère de le voir contourner les complications qui protègent leur corporation. C'est un moment délicat.

Le pire ennemi de l'auditeur, c'est cependant la « tache aveugle » de son propre intellect (voir page 564), la tentation d'éliminer des choses qu'il entend mais qui le contrarient. Les personnes au tempérament impérieux sont incapables d'écouter ; il leur est difficile d'accéder à la pertinence même (et peut-être surtout) si elles sont intellectuellement brillantes.

6.3 Apports de l'informatique à la philosophie

Si la définition et la mise en œuvre du système d'information rencontrent de si grandes difficultés, c'est parce qu'elles impliquent une *innovation philosophique*, un changement de notre façon de penser et de voir le monde de la nature.

Nous sommes les héritiers de la philosophie grecque, qui nous a habitués à voir le monde à travers une grille conceptuelle (Platon, 428-348 ; Plotin, 205-270). Les choses nous apparaissent posées les unes à côté des autres selon leur essence intemporelle : le stylo que je tiens à la main m'apparaît *hic et nunc* comme un objet doté d'une forme géométrique, d'une couleur et d'une fonction précises.

Mais je pourrais le voir aussi sous l'angle de son évolution : la forme de ce stylo résulte d'une conception, il a été produit par une entreprise à partir de matières premières, il a été commercialisé et distribué, je l'ai acheté, je m'en sers, un jour il sera usé et je le jetterai. Cette approche du monde par les *processus* est familière à la pensée chinoise (Confucius, 551-479).

Même si nous savons bien que les choses évoluent (à commencer par nous-mêmes), cette deuxième approche ne nous est pas aussi naturelle, aussi facile qu'elle l'est pour des Chinois : notre pensée spontanée est intemporelle, conceptuelle. Or l'informatique conjugue les deux approches (figure 6.1). Si elle s'appuie sur la grille conceptuelle que définit le référentiel, c'est pour équiper et outiller le flux du processus de production de l'entreprise tout comme celui du processus de traitement des données.

L'informatique nous invite par ailleurs à enrichir notre raisonnement : alors que la démarche qui nous a été enseignée, fondée sur la logique et les mathématiques, procède par déduction à partir de définitions ou d'axiomes,

la programmation informatique nous contraint à modéliser la façon de *faire* les choses : elle ne peut pas se contenter « what is », il faut y introduire la description explicite du « how to » (Abelson et Sussman [197] p. 22).

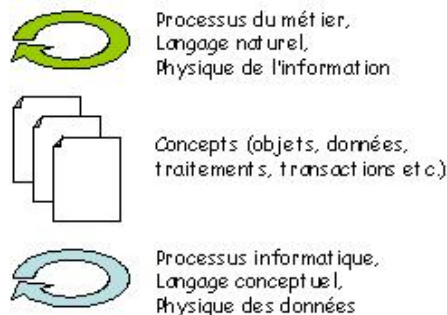


FIG. 6.1 – *Processus métier et processus informatique*

Dans sa conception comme dans sa mise en œuvre, *l'informatique donne ainsi la priorité au processus sur les concepts*. En soumettant les concepts au processus, elle met l'abstraction au service de l'action, ce qui implique une *pratique de l'abstraction*, une maîtrise de la définition et la manipulation des concepts qui constituent la « boîte à outils » du système d'information.

Le processus lui-même se dédouble : au flux des actions sur le monde réel, par lequel se concrétise la vie de l'entreprise, à la *physique de l'information* qui délimite l'action et le comportement des êtres humains, répond dans le système informatique une *physique des données*, le flux qui alimente mémoires et processeurs sous les contraintes de la volumétrie et de l'intégrité. Le processus informatique redouble ainsi le processus de l'action à la façon dont la doublure et la toile redoublent et soutiennent le tissu d'un vêtement ; la conception du système d'information embrasse ces deux processus.

Il n'est pas facile de réfléchir ainsi sur le système d'information. Les informaticiens, même s'ils utilisent couramment l'abstraction, sont loin de mesurer la portée philosophique de leur activité ; les philosophes, pour la plupart, s'intéressent plus à l'histoire de leur discipline qu'à ce qui se passe à leur époque dans un lieu qui se trouve, comme l'entreprise, loin de la chaire du professeur ; les entreprises enfin, prisonnières de trivialité du *business is business*, refusent une réflexion qui leur semble futile. L'informatique occupe ainsi une de ces charnières dont les spécialités se détournent, car il est plus facile de dévaler une pente que de se tenir sur une crête. Elle articule même trois charnières :

- entre l'EHO et l'APU, l'être humain organisé et l'automate programmable doué d'ubiquité ;
- entre concepts et processus, « what is » et « how to », pensée et action ;
- entre la technique et la stratégie.

La technique, c'est étymologiquement le *savoir-faire* (τέχνη). Or le *comment faire* ne peut s'appliquer que si l'on a défini le *pourquoi faire* et le *quoi faire*. Celui qui maîtrise une technique obéit à un but dont la définition

ne relève pas de la technique, même si cette définition doit tenir compte du possible que l'état de l'art circonscrit. Un travail technique, même complexe, et même si sa réalisation se découpe en étapes, est orienté à la fois par la connaissance du but et par celle de l'état de l'art.

À l'opposé de la fonction du technicien, la fonction essentielle du stratège est de gérer l'incertitude. Alors que le processus de production tourne avec la régularité d'un moteur, le stratège se tient au périscope et, vigilant, surveille l'environnement de l'entreprise : cette vigilance nourrit son rôle d'arbitre.

Les techniciens définissent des projets qui visent à mieux faire fonctionner les processus ; le stratège apporte sa valeur ajoutée en arbitrant entre ces projets, voire en suscitant des projets auxquels les techniciens n'auraient pas songé.

La maîtrise d'ouvrage se trouve, aux côtés du stratège, à la charnière des deux mondes de la technique et de la stratégie. Elle assiste le stratège dans ses arbitrages en éclairant le possible technique et l'état de l'art du métier. Elle prépare pour lui un tri des priorités. Elle aide à définir son tableau de bord. Enfin, elle l'aide à prendre la mesure des dimensions stratégiques du système d'information.

6.3.1 Un terrain d'expérimentation

L'informatique est un *terrain d'expérimentation philosophique*. Elle étend en effet la démarche expérimentale, conçue pour explorer le monde de la nature, à l'exploration du monde de la pensée lui-même.

À l'origine de nos systèmes d'information se trouvent trois abstractions :

- choisir, parmi les *êtres* que le monde comporte, ceux qui seront identifiés dans la base de données, en faisant donc abstraction des êtres qui ne seront pas identifiés ;
- choisir, parmi les *attributs* que l'on peut observer sur un être que l'on a identifié, ceux que l'on retiendra pour le décrire dans la base de données, en faisant donc abstraction des attributs qui ne seront pas observés ;
- choisir, parmi les *vues* que l'on peut définir sur la base de données, celles qui seront proposées à tel segment d'utilisateurs, en faisant donc abstraction des vues qui ne seront pas proposées.

Construire un système d'information requiert ainsi une *pratique de l'abstraction* qui met quotidiennement et familièrement en œuvre, et à l'épreuve, les catégories de la pensée. Cela requiert aussi de représenter, lorsqu'on modélise un cycle de vie, le fait qu'un être conserve son identité et reste donc *le même* tout en se transformant : complétant l'abstraction par des hypothèses sur la causalité, c'est là une *pratique de la théorie*. Les abstractions, les théories requises par le système d'information sont au service de *l'action* de l'entreprise sur la nature : ces pratiques ont elles-mêmes une fonction pratique.

Le système d'information permet ainsi d'observer *in vivo* l'articulation entre la pensée et l'action. Il met en scène les démarches de l'abstraction et de la théorie, chaque fois dans un contexte économique, historique et socio-

logique particulier. Articulant enfin l'automate au travail de l'être humain, il invite à explorer leur complémentarité.

Mais dans la culture occidentale la production des concepts et des théories est traditionnellement considérée comme une tâche difficile réservée à des spécialistes, les « penseurs », qui seuls auraient la légitimité nécessaire et qui se contrôlent mutuellement au sein de la « cité scientifique » (Bachelard [7]). Celui qui se conforme à cette tradition risque pourtant, tournant le dos au $\gamma\nu\omega\tau\iota$ $\sigma\alpha\upsilon\tau\acute{o}\nu$ socratique²¹, d'ignorer l'articulation entre sa propre pensée et son action quotidienne.

Il produira bien sûr pour son propre usage des concepts et des théories (puisque c'est ainsi que notre cerveau fonctionne), mais sans le savoir et donc sans contrôler sa propre activité mentale. Il refusera de considérer la production intellectuelle lorsqu'elle est le fait non d'un savant doté de la légitimité académique, mais d'un simple collègue que l'on rencontre chaque jour dans les couloirs de l'entreprise et avec qui l'on déjeune parfois à la cantine.

Ainsi des penseurs méconnus triment dans les soutes de nos DSI et des entreprises spécialisées en informatique. Seuls ceux qui les voient travailler de près peuvent les estimer à leur juste valeur, les autres ne croient pas devoir les respecter. Mais devant un directeur général, un président, un ministre, ah ! comme on sera déférent, que de garde-à-vous...

Je ne sais que penser de ceux qui méprisent le terrain d'expérimentation qu'offre l'informatique en disant « c'est de la technique ». Qu'ils prennent garde à ne pas faire comme ces théologiens qui, au XVII^e siècle, ont refusé de regarder dans la lunette que leur proposait Galilée : cela ne pouvait rien leur apprendre, disaient-ils, puisque tout est déjà dans Aristote et saint Thomas²². Si aujourd'hui un philosophe estime que l'informatique ne peut rien lui apprendre, serait-ce parce qu'il croit que tout est déjà dans les auteurs certes respectables du programme canonique, qu'il s'agisse de Platon ou de Kant, Hegel, Heidegger et autres Derrida ?

* *

Nous avons hérité des Grecs une pensée lumineuse, mère de la philosophie et des mathématiques. La clarté qu'elle projette sur le monde a repoussé tout ce qui n'était pas *pensable* vers l'obscurité du mythe. Mais il se peut que cette clarté nous aveugle. D'autres pensées, moins solidement bâties peut-être mais qui n'ambitionnaient pas avant tout la solidité, apportent à la pensée grecque des compléments et des correctifs précieux.

6.3.2 À l'origine de la pensée occidentale : l'Être

Il n'est pas aisé de distinguer, dans nos perceptions, ce qui reflète authentiquement le réel de ce qui n'est qu'apparence ; ni de distinguer, dans notre pensée, l'image de la réalité de ce qui n'est qu'imaginaire ; ni encore

21. Platon, *Philèbe*.

22. Needham [145] vol. 2 p. 90.

de distinguer, parmi les faits et les êtres, ce qui existe de ce qui n'est que possible.

Il n'est donc pas surprenant que les Grecs, qui les premiers ont exploré le monde de la pensée et qui étaient épris de clarté, aient voulu répondre à la question « qu'est-ce qui *est* vraiment? » ou, de façon équivalente malgré la différence de formulation, « qu'est-ce que *l'être*? ».

À cette question, Parménide (VI^e-V^e siècles) a répondu de façon tranchée : une même chose ne peut pas à la fois être et cesser d'être, ce serait contradictoire. L'être est donc nécessairement immuable. Il en a donné une image suggestive, celle d'une sphère homogène et immobile.

Platon (427-348) est parti de la même intuition : *l'être est immuable*. Mais il l'a libérée de l'image physique à laquelle Parménide avait eu recours et il a délimité avec précision ce qui seul est immuable : ce sont les *Idées*, ou concepts, qui peuplent le monde de la pensée. Et il est vrai que les concepts de cercle, de triangle, de nombre premier etc. sont immuables : si l'on peut définir chacun d'entre eux de plusieurs façons, ses diverses définitions sont équivalentes et donc, à une équivalence près, identiques.

Cependant si seuls les concepts possèdent l'être, si seul est réel ce qui est immuable et susceptible d'être défini, ni vous ni moi ne sommes réels puisque nous sommes nés un jour, que nous ne cessons d'évoluer, qu'un jour nous mourrons et qu'il serait vain de tenter de nous définir. Platon, parfaitement cohérent, refuse de dire que nous sommes réels : les êtres humains, les animaux, les plantes, le monde de la nature tout entier ne sont réels, selon lui, que dans le concept sous lequel on peut les ranger. Le cheval qui est là dans le pré n'est qu'une apparence, l'être réside dans le concept de cheval ; de même, vous et moi ne sommes que des apparences, l'être réside dans le concept d'être humain.

Il établit ainsi une cloison étanche entre le monde de l'expérience, dans lequel il ne voit qu'une illusion, et le monde des Idées que seul il estime réel. C'est ce qu'exprime, dans *La République*, le mythe de la caverne. Mais un tel système ne peut convenir qu'à ceux qui, vivant dans le monde de la pensée, préfèrent le préserver de tout contact avec le monde de la nature.

* *

Aristote (384-322) s'intéressait passionnément au monde de la nature et son intuition s'est révoltée contre celle de Platon (Gilson [68]). Non, a-t-il dit, ce ne sont pas les Idées qui sont réelles mais les *choses*, considérées individuellement, une par une et avant toute intervention de la pensée. Il confère ainsi l'être à des choses qui ne sont pas immuables : à vous, à moi, aux animaux, aux plantes, aux minéraux etc. Il rejoint ainsi heureusement le sens commun dont Platon s'était si délibérément écarté.

Mais après avoir reconnu l'existence d'une chose Aristote cherche à dire ce que cette chose *est*, à décrire sa *forme* ou encore son *essence*. Et l'essence d'une chose, dit-il, c'est ce qui est *pensable* en elle, sa représentation dans la pensée.

Ainsi, après avoir placé l'être dans l'individu existant, il réduit l'individu à ce que l'on peut penser de lui. Puis il réduit encore ce pensable à une

catégorie, ou prédicat, sous laquelle il classe l'individu : « τὰὐτὸ γὰρ εἷς ἄνθρωπος καὶ ὢν ἄνθρωπος καὶ ἄνθρωπος » : « un homme », « un homme existant » ou « homme », dit-il, c'est tout un²³. Aristote est ainsi tout aussi idéaliste que Platon, quoique d'une façon différente. La chose individuelle, point de départ de son intuition, se résorbe dans une essence, puis cette essence se résorbe dans un classement. Certes l'essence d'une chose ne saute pas aux yeux : ce que la chose a d'*essentiel* lui est aussi intime qu'un secret ; mais le but de l'effort de connaissance est de le dégager.

Aristote ne mentionne cependant pas qu'une même chose puisse avoir dans la pensée des représentations diverses selon le point de vue à partir duquel on la considère. L'essence d'une chose est selon lui unique et il suffit de la connaître pour penser la chose de façon adéquate. Ce postulat est nous le verrons invalidé par l'expérience mais il a été repris tel quel par des philosophes qui, à la suite d'Aristote, ont préféré déduire alors qu'il aurait fallu observer.

* *

Les Grecs ont, les premiers, arpenté le monde de la pensée : ils ont les premiers découvert la puissance de l'abstraction. Il n'est pas surprenant qu'ils se soient, comme le fait tout inventeur, exagéré la portée de leur découverte. Il était sans doute inévitable qu'ils surestiment la capacité de la pensée à rendre compte du monde.

L'énergie qui se dégage de leurs écrits a séduit tous ceux qui, après eux, ont entrepris de réfléchir. Il en est résulté des habitudes qui se sont enracinées dans nos procédés de pensée. On dit ainsi, par exemple, qu'un scientifique doit être *objectif* : cette expression ne signifie pas seulement qu'il convient d'être intellectuellement honnête, car cela va sans dire ; elle signifie que la pensée doit reproduire l'objet tel qu'il est, sans que sa connaissance ne dépende en rien du sujet qui connaît.

Cela se conçoit dans le système d'Aristote : pour que la pensée puisse atteindre l'essence de l'objet qu'elle vise, il faut qu'elle se focalise sur lui en faisant abstraction du point de vue de l'observateur. Mais si l'on admet qu'un même objet puisse être considéré à partir de divers points de vue à chacun desquels correspond une représentation spécifique, alors il faut indiquer, avant de dire comment on le représente, le point de vue à partir duquel on l'a considéré - ce qui est *subjectif*, même s'il ne s'agit pas d'une subjectivité individuelle mais de celle d'un point de vue, et même si le choix de ce point de vue peut objectivement correspondre à la situation de l'observateur.

6.3.3 Opacité de l'existant

Thomas d'Aquin (1225-1274) se trouve à l'articulation des pensées grecque et juive. Il récupère l'héritage scientifique d'Aristote mais se sépare de lui par la distinction entre existence et essence (Gilson [67]).

23. Aristote, *Métaphysique*.

Chez Aristote, une fois que la pensée a atteint l'essence d'une chose, elle peut se dispenser de considérer sa génération et sa corruption (en d'autres termes, sa naissance et sa mort); elle peut donc se dispenser de considérer l'origine du monde.

Mais Thomas d'Aquin, héritier de la Bible, ne pouvait pas ignorer la création. Il sépare alors par un trait bien net l'existence de l'essence. L'existence, c'est l'acte d'être, brut et avant toute qualification: un existant se propose à la pensée comme objet, mais elle ne saurait rendre compte du fait qu'il existe car ce fait est antérieur à la perception comme à la réflexion.

Bien plus: aucune pensée, aucune essence ne pouvant rendre intégralement compte d'un existant, *tout existant est opaque à la pensée*. Chaque existant est un mystère. Il en est de même de Dieu, l'Existant même, dont émane toute existence et qui est lui aussi inconnaissable.

Il y a là, pour ceux qui s'étaient habitués à ramener chaque existant à son essence, puis à raisonner sur lui à partir d'elle, quelque chose de désespérant. La pensée de Thomas d'Aquin révolte en nous non pas le sens commun - auquel elle adhère exactement - mais des habitudes acquises à l'école, formées par l'école, et qui sont peut-être pour la pensée un mauvais pli.

Oubliez l'école et regardez en effet les êtres qui vous entourent. Ils *existent*, c'est là un fait brut à partir duquel votre pensée peut se mettre à l'œuvre mais qui lui est antérieur, extérieur, et qu'elle ne peut pas expliquer. Regardez vous dans un miroir: vous y voyez un primate évolué, doté d'un corps qui fonctionne sans que vous ne l'avez voulu ni pensé et qui fixe sur vous un regard énigmatique. Regardez vos mains: sont-elles « pensables »?

Regardez cette plante avec ses nervures, ses canaux, ses cellules, sa composition chimique, et aussi son passé et son avenir: votre pensée peut-elle rendre compte de son existence? Peut-elle la représenter de façon exhaustive, parfaite, complète, absolue?

Regardez le système d'information de votre entreprise. Il contient une base de données sur les clients. Quels sont les attributs qu'elle retient pour décrire un client? Son nom, son adresse, son numéro de téléphone, le nom de son entreprise, peut-être. Mais notez vous son poids, sa taille? Oui si vous êtes son médecin, non sans doute si vous êtes son libraire. Notez vous la couleur de ses yeux? Oui si vous êtes le policier qui remplit une fiche signalétique, non si vous êtes un boulanger ou un postier. Notez vous le nombre de ses cheveux? Non, car ce nombre change tout le temps; pourtant à chaque instant il a une valeur précise...

Le fait est que ce que nous voyons, ce que nous observons, ce ne sont pas des essences qui rendraient compte chacune d'un des objets que nous considérons, mais des *vues* partielles et *choisies*. Que l'on puisse, que l'on doive considérer un objet selon le point de vue qui correspond à la relation que l'on a avec lui, que du coup un même objet puisse être considéré selon divers point de vue par des personnes qui ne se trouvent pas dans la même situation à son égard, que l'unicité de l'essence éclate ainsi en autant de représentations qu'il existe de points de vue, c'est là un fait que l'expérience constate.

Tout cela peut paraître compliqué et en effet il serait plus « simple » de supposer que l'on puisse associer à chaque objet une essence et une seule : seulement cela ne marche pas. Supposez qu'une équipe soit chargée de définir le référentiel d'une entreprise, la grille conceptuelle qu'elle va utiliser pour décrire les êtres avec lesquels l'entreprise est en relation. Si cette équipe entreprend de décrire l'essence de ces êtres, elle s'engage dans une tâche sans fin car elle ne dispose d'aucun critère formel qui permette de distinguer ce qui est important de ce qui ne l'est pas, ni de définir le degré de détail, le « grain de la photo », auquel il convient de s'arrêter. J'ai vu des équipes travailler de la sorte pendant des années sans produire quoi que ce soit d'utilisable.

Pour que tout s'éclaire, il suffit de dire « que s'agit-il de *faire* ? » : alors il devient possible de distinguer l'essentiel de l'accessoire, de faire abstraction des aspects *dont on n'a que faire*, de concevoir le degré de détail raisonnable.

Il n'est certes pas toujours facile de définir ce que l'on fait, ce que l'on produit : que produit une direction des achats ? une direction des ressources humaines ? l'état-major des armées ? un institut statistique ? On accordera que s'il est possible de travailler par habitude et sans savoir ce que l'on produit, il est préférable d'avoir tiré cette question au clair : on risque sinon d'agir à rebours d'une mission que l'on ignore.

Les théories dites « de la complexité » tâtonnent à la rencontre de ce fait : le nœud de la complexité, c'est l'opacité de l'existant, la diversité sans limite des points de vue que l'on peut légitimement prendre sur lui, la diversité des représentations qui en résultent. Mais souvent les tenants de ces théories croient trouver la complexité dans des procédés de pensée ou dans des mécanismes comme l'articulation de plusieurs logiques, que formalise le modèle en couches ; le croisement de plusieurs codages, plusieurs classifications (voir page 206) ; la rétroaction (*feedback*). Ils ne font ainsi que suivre la pente sur laquelle les Grecs ont lancé la philosophie : si seul le pensable est réel, on doit pouvoir atteindre l'existant dans la pensée même. C'est ce qu'ont tenté Hegel avec la dialectique, Bergson avec la durée. La philosophie répugne, malgré Thomas d'Aquin, Pascal [152] et Kierkegaard [102], à admettre l'opacité de l'existant.

* *

Les penseurs de la Renaissance avaient redécouvert la philosophie grecque, dont ils héritèrent le goût pour la pensée claire et explicite. Ils lui adjoignirent le goût pour l'observation : le couple ainsi formé donnera naissance, avec Galilée (1564-1642), à la démarche expérimentale et à la science occidentale.

Mais à la même époque Boileau écrivit un vers que l'on cite avec trop de complaisance : « Ce qui se conçoit bien s'énonce clairement » (Boileau [17]). C'est là une contre-vérité manifeste. Vous vous représentez clairement le visage de l'être aimé, vous le reconnaissez entre des millions d'autres, mais vous êtes incapable de le décrire car il est impossible de décrire un visage avec des mots (une photographie ferait l'affaire, mais elle ne « s'énonce » pas). Autres exemples : le général doué du « coup d'œil » sait concevoir la manœuvre opportune, le cuisinier de talent réussit ses plats, le champion

motocycliste choisit la meilleure trajectoire - mais ils sont incapables de dire comment ils s'y sont pris. Beaucoup des opérations de notre pensée nous sont aussi obscures que le fonctionnement de nos organes. Cela ne veut pas dire qu'elle fonctionne mal, ni que ses résultats soient fallacieux, même si nous ne sommes pas en mesure de les expliciter.

En héritier de la Renaissance Boileau dévalue la pensée implicite : ne peut avoir été bien conçu, dit-il, que ce que l'on sait énoncer. C'est que les hommes du XVII^e siècle se sont détournés des épisodes obscurs qui précèdent l'expérimentation ; ils n'ont pas voulu voir l'entre-deux où l'esprit flotte pour anticiper sur la déduction et choisir, dans l'illimité du possible logique, les axiomes qui pourront être les plus féconds. L'âge classique n'a voulu connaître de la science que ses *résultats*, présentés formellement et selon la stricte rigueur, et *il a préféré ignorer sa démarche*. Or celle-ci accorde, dans la phase initiale qui l'amorce, une large place aux associations d'idées, aux analogies, aux considérations esthétiques (Poincaré [162]).

Les pédagogues présentent le savoir sous forme de définitions et de déductions. Quelqu'un qui a une bonne mémoire et un esprit clair peut aller loin dans les études sans jamais s'être examiné lui-même (ce qui serait pourtant le *minimum minimorum* de la démarche expérimentale), sans avoir observé le monde de la nature, sans avoir entrevu ce qui faisait la vie des chercheurs ni l'intention des recherches dont il a absorbé les résultats.

6.3.4 De la pensée à l'action

Lorsque Emmanuel Kant (1724-1804) a établi que la pensée ne pouvait pas s'égaliser au réel, cela a désespéré certaines personnes au point qu'elles se sont suicidées, comme Heinrich von Kleist (1777-1811). À quoi bon penser, à quoi bon vivre, se sont-elles dit, si ma pensée ne peut pas atteindre l'Absolu !

Si elles avaient perçu la finalité *pratique* de la pensée, si elles s'étaient libérées du mirage de la connaissance absolue, elles n'auraient pas connu ce désespoir. Lorsque je considère un objet, la grille conceptuelle à travers laquelle je le perçois et le décris est-elle la bonne, sachant qu'il existe *a priori* une infinité de grilles formellement correctes et toutes également possibles ? Pour en décider, je n'ai pas d'autre critère que celui de la *pertinence*, de l'adéquation à l'*action* que j'entends mener. C'est ce qu'illustre l'analyse historique des classifications et tables de codage : les auteurs des nomenclatures selon lesquelles on classe les produits, les activités économiques, les classes sociales etc. ont tous prétendu produire la nomenclature « naturelle » mais ils ont utilisé pour cela des critères d'agrégation qui répondaient aux besoins de l'économie ou de la société de leur temps et qui, comme ces besoins, ont évolué (Guibert, Laganier et Volle [77]).

On voit dès lors s'évanouir l'ambition d'une connaissance qui reproduirait l'objet indépendamment de l'action qui le vise, de l'*intention* : au contraire, c'est l'intention qui fournit le critère selon lequel on pourra évaluer la représentation. En même temps on saura que la représentation qui répond à ce critère peut ne pas convenir à une autre intention, donc qu'elle n'est pas l'essence unique de l'objet.

Cela ne veut pas dire que nous soyons libres d'observer ni de penser n'importe quoi : on retrouve les exigences de l'objectivité mais sous une forme plus élaborée que celle, vraiment sommaire, qui prétendait reproduire l'objet dans la pensée. Le choix des concepts pertinents n'a rien d'arbitraire, ni l'observation que l'on fait à travers la grille conceptuelle choisie. Lorsque je conduis ma voiture, il est *nécessaire* que j'utilise la grille adéquate où figure, entre autres, le concept de feu avec ses trois modalités ; le fait que ce feu devant moi soit vert, rouge ou orange ne dépend pas de ma fantaisie. Le monde se reflète dans une telle grille de façon certes incomplète, mais *authentique*. Le caractère incomplet de la grille est d'ailleurs favorable à l'action car il focalise l'attention sur les seuls éléments que celle-ci doit considérer.

Relier la pensée à l'action dénoue l'angoisse que suscite l'opacité de l'existant. Si en effet la finalité de la pensée est essentiellement pratique, peu importe qu'elle ne puisse pas nous restituer l'existant dans l'absolu : il suffit qu'elle nous procure les moyens d'agir sur lui avec justesse. Tout existant étant pour notre action à la fois un obstacle et un outil, sa représentation dans notre pensée n'a pas d'autre but que de nous fournir les poignées mentales qui nous donneront *prise* sur lui, qui nous permettront de le *manipuler*.

* *

Mais si l'on évalue la représentation selon son adéquation avec l'action, il reste à évaluer l'action elle-même : est-elle judicieuse ou non ?

Elle le sera si elle est en accord avec l'*intention*, si l'on fait effectivement ce que l'on a la volonté de faire, ce qui suppose que l'on ait tiré l'intention au clair et qu'on l'ait dégagée du conflit intime que se livrent en nous des intentions simultanées mais inconciliables : on ne peut pas vouloir à la fois *être* et *paraître* ; on ne peut pas vouloir à la fois la justice et l'arbitraire etc.

Mais les intentions elles-mêmes, comment les évaluer ? Il faudra les rapporter aux *valeurs* auxquelles on adhère et qui, pour nous, sont sacrées en ce sens que nous sommes prêts à leur *consacrer* notre vie et, s'il le fallait, à la leur *sacrifier*. Ces valeurs sont le ressort de nos intentions ; elles fondent la *volonté voulante* qui anime notre volonté voulue et explicite. Cependant le plus souvent elles échappent à notre pensée, elles nous animent sans que nous puissions les expliciter.

Beaucoup de personnes haussent les épaules lorsqu'elles entendent le mot *valeur* qui, disent-elles, « ne veut rien dire ». Elles n'ont peut-être pas examiné avec assez d'attention leur propre fonctionnement intime. Il est vrai que cela marche tout seul, tout comme l'estomac digère sans que l'on n'y pense ; mais cela n'en est pas moins opératoire et efficace. Cela peut aussi être sujet à des pathologies : alors cela fait mal et on se rend compte que c'est réel, tout comme on sent l'existence de son estomac quand il a un ulcère.

Une des tâches les plus importantes de la réflexion, c'est de tirer au clair l'écheveau des valeurs qu'impliquent nos intentions, que révèlent nos réflexes, pour en chasser les incohérences : car si notre cœur est le théâtre de valeurs incompatibles (« il faut être discipliné et obéissant, tout en étant

original et intraitable »), nos intentions seront désordonnées et nous tournerons dans le cercle de l'activisme, l'action d'un jour annulant celle de la veille.

* *

La pensée occidentale a subi à la Renaissance une coupure qui l'a mutilée en même temps qu'elle la fécondait. La démarche expérimentale, l'audace devant un monde que la pensée explore librement, ont ouvert la voie au déploiement de la science et des techniques. Elles ont polémique à bon droit contre l'argument d'autorité, le dogmatisme, et contre certains procédés de pensée qui tournaient à vide. Mais elles ont rejeté aussi les techniques antiques de la mémoire et, plus généralement, de la pensée symbolique (Yates [221]). En nous coupant ainsi de l'histoire de la pensée, la Renaissance a donné naissance à de nouvelles formes de dogmatisme et de pédantisme : le rationalisme n'a pas toujours été raisonnable.

La pensée symbolique procède par analogies, associations d'idées, et résiste à l'explicitation. Elle ne cherche pas à énoncer, mais à suggérer ; elle sollicite une interprétation qui, le plus souvent, ne peut pas être univoque. Et pourtant la suggestion sera, dans la communication entre des êtres humains, souvent mieux comprise qu'un énoncé explicite. L'étymologie du mot « symbole » ne renvoie pas vers « imaginaire » mais, de façon plus profonde, vers le nœud qui relie différentes choses : *συμβολή* veut dire jonction, réunion, rencontre. Nous allons, pour illustrer cela, partir du rêve pour aller jusqu'aux bases de données et à la connaissance.

Georg Groddeck (1866-1934), dans *Le livre du Ça* [75] (1923), a critiqué l'interprétation des rêves par Sigmund Freud. Freud ne donne en effet qu'une seule interprétation d'un même rêve ; Groddeck par contre les multiplie, toutes différentes et toutes également plausibles. Le rêve, comme symbole, est le nœud qui réunit ses diverses interprétations ; par delà le sens explicite de chacune d'elles il pointe vers un sens implicite qui leur est commun, mais que des paroles ne pourraient pas exprimer.

Il en est de même pour la Bible. Ce texte, antérieur à la formation de la pensée conceptuelle, est symbolique et puissamment suggestif. On peut l'interpréter de diverses façons qui toutes pointent vers un sens que des mots ne sauraient exprimer : aucun commentaire ne peut l'épuiser et le pire des contresens peut résulter d'une lecture qui prendrait le texte à la lettre.

Lorsque nous réfléchissons, dans la phase exploratoire et rêveuse qui précède la formation des concepts, l'esprit flotte au gré des associations d'idées que notre mémoire alimente, que nos procédés de pensée activent ou que la glande cérébrale sécrète spontanément ; des ébauches de déduction s'esquissent à partir de définitions à peine posées, sitôt rejetées ; des images se projettent sur un écran intérieur, des personnages y jouent des scènes hypothétiques, des architectures se créent et se dissipent. Voici que l'une d'entre elles prend corps, s'organise : nous saisissons un papier pour la noter en quelques phrases, puis nous laissons de nouveau notre esprit flotter d'une image à l'autre, d'un symbole à l'autre, soucieux d'éprouver la solidité de la structure que nous venons d'entrevoir et désireux d'en ramener d'autres, si possible, dans nos filets (Colwell [39] p. 5).

Lorsque nous voulons dire à quelqu'un d'autre ce que nous pensons, il serait vain de chercher à nous expliquer entièrement : l'interlocuteur serait noyé sous un flot de paroles. Mieux vaut user de quelques images suggestives qui vont l'inviter à partager notre intuition et à faire le même parcours que nous, pour enfin pouvoir se représenter ce que nous avons en tête. Cela ne marche pas toujours, mais c'est seulement ainsi que cela peut marcher.

Une base de données est invisible : il est impossible de l'afficher en entier sur un écran, de l'imprimer en entier dans un document - et le serait-ce que ce document serait illisible. Mais on peut - *et cela suffit* - donner à chaque utilisateur sur cette base la « vue » qui répond à ses besoins. Les diverses vues sont toutes différentes mais ce qui fait leur unité, c'est qu'elles se réfèrent toutes à la même base, que celle-ci les rassemble comme un nœud : la base de données est un symbole !

Tout objet concret, existant, se présente à nous comme un nœud qui rassemble un nombre illimité de représentations possibles parmi lesquelles nous devons choisir en fonction de nos besoins pratiques. Nos concepts ne nous en donnent que des vues partielles ; ce qui fait l'unité de ces vues, leur cohérence, c'est qu'elles se réfèrent toutes au même objet : un même objet ne peut pas être en même temps, et sous le même rapport, à la fois une chose et son contraire. Ainsi, chaque objet concret, existant, est lui aussi un symbole !

Cette dernière phrase peut surprendre. Mais il existe une pensée avant que les concepts ne soient construits, une pensée qui a pour tâche de *choisir* les concepts. Et avant de choisir les êtres que l'on va observer il faut avoir conscience du monde de la nature, tout comme il faut avoir conscience du monde de la pensée avant de choisir les axiomes d'une théorie. Nous avons de tout existant une conscience préconceptuelle qui le considère tel quel, avec ses attributs innombrables et encore innommés. Cette conscience antérieure au concept, antérieure à la pensée construite et opératoire, antérieure à toute détermination, cette conscience rêveuse et flottante - mais confrontée de façon immédiate à la consistance de l'existant - relève tout entière de la pensée symbolique.

6.3.5 De l'action aux valeurs

Que la pensée ait une finalité pratique, qu'elle vise l'action, c'est un fait dont les cabalistes se sont avisés depuis longtemps (Steinsaltz [194], Mopsik [141]). Ils distinguent quatre mondes : (1) le monde de l'action, dont relève la pensée elle-même ; (2) le monde de la formation ; (3) le monde de la création ; (4) le monde de l'émanation. On peut, en interprétant ce modèle en couches, dire que l'intention (qui motive l'action) relève du monde de la formation, que les valeurs (qui orientent les intentions) relèvent du monde de la création, et que le monde de l'émanation confine à l'infini (*En-Sof*).

« Ἐγώ εἰμι ἡ ὁδὸς καὶ ἡ ἀλήθεια καὶ ἡ ζωὴ » (Jean 14:6) : « Je suis le chemin, la vérité et la vie ». Dans cette phrase on peut être attentif à l'ordre des mots, et voici une interprétation possible. Le chemin est mentionné en premier : on ne doit pas se reposer sur ce que l'on possède, il faut avancer. La

vérité ne vient qu'en second : plutôt qu'un bien que l'on pourrait tenir dans sa main, elle est à l'infini de l'horizon comme un point lumineux qui oriente le chemin mais semble reculer à mesure que l'on avance vers lui. Enfin vivre n'est rien d'autre que de suivre fidèlement (*fides*) le chemin ainsi orienté.

Il se peut que la dogmatique ait oublié cette conception modeste de la vérité. Karl Popper l'a redécouverte au cœur même de la science (Popper [163]) : une théorie scientifique ne peut pas être « vraie » au sens où peut l'être l'énoncé d'un fait car elle suppose une induction qui, généralisant une observation inévitablement limitée, pourrait être invalidée par une expérience ultérieure. La scientificité d'une théorie s'évalue non seulement par le fait qu'elle n'a pas été contredite par les expériences connues, mais aussi par le fait qu'elle est construite de façon à être vulnérable (« falsifiable ») par l'expérience future. Ainsi les théories construites de façon à interdire toute réfutation, et que l'on pourrait croire définitives, sont non scientifiques en raison même de leur solidité apparente.

La connaissance apparaît alors comme une zone lumineuse qui peut s'élargir, mais se découpe sur un plan infini qu'elle n'éclairera jamais en entier et dont elle ne couvre donc qu'une part infime, aussi imposantes que soient ses constructions.

La pensée chinoise accorde elle aussi la priorité à l'orientation, au *chemin* (dào, prononcer tao). Pour les Classiques chinois les définitions ne décrivent pas l'essence des choses et ils ne s'y intéressent pas d'un point de vue abstrait. Elles ne sont pour eux que des instruments en vue du contrôle de l'environnement physique et social. Ils prisent donc moins l'ingéniosité des définitions que le *discernement* qui permet d'établir des distinctions utiles. Pour eux, un concept n'a de valeur que s'il a une utilité pratique, s'il est pertinent, et la parole a moins de valeur que l'action (Elisseff [53]) : « quand la Voie règne, dit Confucius, l'action fleurit ; quand la Voie ne règne pas, c'est la parole qui fleurit ». Les mots sont des pointeurs vers une réalité qu'ils n'atteignent pas : « viser n'est pas atteindre », (zhi bù zhì, prononcer djeu pou djeu)(Hui Shih, cité par Needham [145] vol. VII:1 p. 49).

Ainsi diverses sagesses montrent notre vie orientée par des valeurs qui déterminent nos intentions et se concrétisent dans notre action. Ces valeurs rencontrent le monde tel qu'il existe : il nous revient de les y manifester pratiquement et symboliquement, de les y *incarner*. Ce monde, notre pensée ne nous permet pas de le connaître exhaustivement, mais elle est pour l'action un outil efficace et *cela doit nous suffire*.

Cheminer vers l'infini alors que notre vie est limitée par le temps et l'espace, comme par l'envergure de l'expérience possible, cela suscite une souffrance qui est inséparable du destin humain. Confucius a mis la compassion (*cum patire*, « souffrir avec », en chinois rén, prononcer jen) au premier rang des valeurs humaines.

Les nazis ont vu par contre dans la compassion une faiblesse : poussant à l'extrême l'injonction *Du mußt hart sein*, « tu dois être dur », ils se faisaient un devoir d'être *erbarmungslos*, impitoyables. Refuser l'humanité leur semblait être un signe de force mais leur brutalité était plutôt le symptôme d'une infirmité. Celle-ci fut présente tout autant chez les staliniens (Mon-

tefiore [139]). Elle perdure aujourd'hui dans la vogue d'un anti-humanisme esthétisant à prétention théorique dont on ne perçoit sans doute pas assez clairement les implications pratiques.

On retrouve la compassion dans l'expression « Ἀγαπᾶτε ἀλλήλους » (Jean 13:34) que l'on traduit par « aimez-vous les uns les autres ». Dans l'entreprise, la nécessité du dialogue entre spécialités différentes comme avec les clients invite à la traduire par « respectez-vous les uns les autres », c'est-à-dire « faites un effort sincère pour comprendre ce que l'autre vous dit » : sans cet effort, il sera impossible de faire coopérer des personnes qui ont des vues différentes sur les êtres avec lesquels l'entreprise est en relation, qui utilisent des grilles conceptuelles différentes et parlent selon des vocabulaires différents. Or il est bon que dans l'entreprise chacun soit assez polyglotte pour pouvoir comprendre, sinon parler, le langage des autres spécialités.

Point n'est besoin, pour fonder l'humanisme, d'avoir recours aux émotions douteuses qui entourent les bons sentiments : l'observation, la simple et ferme logique y suffisent. Ce que chacun possède de plus précieux et de véritablement sacré, par delà ses particularités individuelles, c'est son humanité même : et nous la possédons tous également. C'est même sous ce seul rapport - mais il est fondamental - que l'on peut dire que les êtres humains sont tous égaux.

La phrase *nous sommes tous des êtres humains* procure alors à l'édifice des valeurs, des intentions et de l'action un fondement aussi simple et aussi solide que le lien que Descartes a instauré entre la pensée et l'existence en disant « je pense, donc je suis » [46].

Troisième partie

L'informatisation de
l'entreprise

Chapitre 7

Socle sémantique

7.1 Langage et « langage »

Selon Saussure [179] une langue est un « système de signes », chaque signe étant un doublet constitué d'un signifiant phonétique et d'un signifié conceptuel¹. Saussure estime impossible de dire lequel, du signifiant ou du signifié, est premier dans le signe ; pour décrire la formation des signes il a recours à une image suggestive ([179] p. 156) : « La pensée, chaotique de sa nature, est forcée de se préciser en se décomposant. Il n'y a donc ni matérialisation des pensées, ni spiritualisation des sons, mais [...] la “pensée-son” implique des divisions et la langue élabore ses unités en se constituant entre deux masses amorphes. Qu'on se représente l'air en contact avec une nappe d'eau : si la pression atmosphérique change, la surface de l'eau se décompose en une série de divisions, c'est-à-dire de vagues ; ce sont ces ondulations qui donneront une idée de [...] l'accouplement de la pensée avec la matière phonique ».

Cette intuition puissante tourne le dos à la conception qui ne voit dans la langue qu'un habillage de concepts préexistants : la création des signes est *à la fois*, et inséparablement, création des concepts et création de la notation sonore qui conditionne leur identification, leur recyclage dans le raisonnement ou l'imaginaire, leur communication.

Les structuralistes, soucieux de mettre à jour ce qui peut être dit dans une langue existante, ne se sont intéressés ni à la création ni à l'évolution de la langue que Saussure avait pourtant évoquées : voulant souligner les contraintes qu'impose l'architecture existante, il n'ont pas cru opportun de faire apparaître les procédés qui permettraient de s'en affranchir ou tout au moins de les déplacer.

Par ailleurs Saussure n'accorde pas de place aux *connotations*, ces liaisons entre signes qui confèrent au langage sa puissance suggestive ; il ne mentionne pas les associations d'idées qui, appuyées sur le réseau des conno-

1. Le terme « langage », lui, associe « langue » et « parole », cette dernière désignant les aspects physiologiques et acoustiques de l'élocution et de l'audition.

tations, sont si fécondes pour la réflexion et si efficaces pour la communication.

Lorsqu'on les considère, on rencontre un fait qui semble contredire Saussure : il arrive, lorsque l'esprit suit le cours des associations d'idées et des raisonnements, qu'il identifie un concept pour lequel il lui faut trouver un nom : dans ce cas la création du signifié est bien antérieure à celle du signifiant.

Les informaticiens de profession connaissent la différence entre le langage de programmation et le langage naturel. Ils savent que le langage de programmation ne doit pas comporter de connotations : l'automate ne pouvant pas interpréter les suggestions, il faut lui donner des consignes parfaitement explicites. La vie courante a par ailleurs tôt fait de leur enseigner que, dans la conversation entre des êtres humains, l'explicitation parfaite qu'exige la programmation n'est pas de mise.

On doit en fait distinguer trois types de langage :

1) le *langage naturel*, connoté, suggestif, alimente une pensée qui fonctionne par association d'idées ; il est exploratoire et créatif, mais au risque d'une erreur logique (contradiction interne) ou d'une inadéquation à l'action ;

2) le *langage théorique*, fondé sur des définitions dont toute connotation doit être écartée, procure la précision nécessaire à l'action. Aux concepts ainsi dégagés il associe des hypothèses causales.

Nota Bene 1 : Les théoriciens attribuent au langage théorique le monopole de la rigueur. Cependant le bon usage du langage naturel est, dans son ordre, aussi exigeant que la construction théorique. Il n'est pas facile en effet de maîtriser la puissance allusive de la langue. C'est là tout l'art de la poésie, et ceux qui se moquent des poètes auraient du mal à égaler le pouvoir suggestif que La Fontaine atteint sans effort apparent :

*Un jour, sur ses longs pieds, allait, je ne sais où,
Le héron au long bec emmanché d'un long cou*².

Nota Bene 2 : Les mathématiques sont le langage théorique pur ; elles explorent, à partir des définitions ou axiomes (« what is »), les implications du principe de non-contradiction et préparent ainsi à l'action en général.

Nota Bene 3 : Associer la théorie à l'action peut surprendre. Cependant il faut bien, pour agir, préciser sur quoi l'on va agir et poser des hypothèses sur les causalités que l'action fera jouer. Le fait que cela soit souvent implicite n'enlève rien à cette opération, mais il faut distinguer la théorie implicite, à l'œuvre, de la théorie explicite qui seule se prête à la communication et à la discussion scientifiques.

Nota Bene 4 : La théorie, qui ne doit comporter aucune connotation, s'élabore en polémiquant avec le langage naturel : si les associations d'idées jouent un rôle essentiel dans la réflexion qui précède et motive l'élaboration théorique, elles ne doivent plus figurer dans l'expression même de la théorie.

2. Jean de La Fontaine (1621-1695), « Le Héron », *Fables*, Livre VII Fable 4.

Il reste cependant nécessaire de fournir, sous forme de commentaires, la clé des intuitions qui ont guidé la construction théorique.

3) le *langage de programmation* vise à faire faire par l'automate un ensemble bien défini d'opérations. On est là dans le monde du « how to », du « savoir-faire », de la mise en pratique. Ce langage doit respecter les contraintes de la cohérence car dans le domaine de l'action le principe de non-contradiction s'applique absolument. Il doit respecter également les contraintes physiques de l'automate (taille des mémoires, puissance des processeurs, débit des réseaux). Les données sont comme dans le langage théorique définies par des concepts purs, sans connotations.

Pour pouvoir formuler les commandes adressées à l'automate, le langage de programmation fournit une syntaxe, c'est-à-dire (1) des règles d'écriture permettant de distinguer programme et commentaire, donnée et fonction, divers types parmi les données, et aussi (2) des « formes spéciales » conventionnelles et des fonctions préprogrammées. *Il ne fournit pas* - mais il est prêt à les accueillir - les tables de codage qui associent à chaque donnée sa définition selon le langage conceptuel de l'utilisateur.

Un langage théorique est plus pauvre que le langage naturel - il est rare que son vocabulaire comporte plus de quelques dizaines de termes - mais il va plus loin dans l'exploration des relations causales entre concepts et de leurs implications. Son formalisme garde cependant peu de traces des intentions qui ont suscité la construction de la théorie, ce qui dérouté souvent l'intuition du non expert.

Un programme informatique est encore plus éloigné de cette intuition parce qu'aux contraintes purement logiques, que l'on comprend aisément après un apprentissage, il ajoute les conventions qui résultent des contraintes de l'automate et dont la compréhension demande une expertise peu courante.

Ainsi, le langage naturel *facilite la communication* entre les êtres humains grâce aux allusions et suggestions qui lui permettent, fût-ce au prix d'une imprécision, de transmettre plus d'information que n'en contiennent ses termes pris à la lettre. Le langage théorique par contre est fait pour *penser l'action* dont il désigne les objets avec précision ; mais étant peu suggestif il se prête plus à la collaboration dans l'action qu'à la communication avec ceux qui sont loin de l'action.

Le langage de programmation est, lui, adapté pour commander l'automate qui va assister l'action humaine. Il n'est pas conçu pour être lu ni compris par un être humain, mais pour être pratiquement efficace³.

* *

En informatique, le mot « logique » désigne non pas les règles formelles dont le respect garantit l'exactitude du raisonnement, mais les règles syntaxiques auxquelles doit se conformer tout langage de programmation. Le

3. Certains auteurs considèrent la lisibilité comme la première qualité d'un programme. Certes un programme bien écrit sera plus facile à corriger s'il en est besoin, mais le principal critère de qualité est bien l'efficacité pratique du programme et non sa lisibilité.

mot « sémantique », qui désigne en linguistique le sens d'un texte pour le distinguer de sa forme, désigne en informatique l'interprétation d'un langage sous forme de structures mathématiques (typage des données etc.).

La normalisation du vocabulaire élague les synonymes, élimine les homonymes et clarifie les concepts. Mais parfois elle grave de faux amis dans le marbre. Celui qui s'efforce d'améliorer le vocabulaire se fait alors reprocher de s'écarter du langage standard ; mais que faire si ce langage est dans l'entreprise source de contresens répétés ? Pour quelques mots bien bâtis, comme « informatique » ou « logiciel », combien de monstres a confortés la normalisation ? Quelle qualité linguistique peuvent d'ailleurs avoir des compromis établis par des comités où s'affrontent les intérêts économiques des industriels ? N'est-il pas trop courant que l'on prétende imposer à tous, sous prétexte de rigueur, le particularisme d'un jargon professionnel ?

L'un des faux amis les plus révélateurs est le mot « donnée », d'utilisation si courante. La « donnée » est la matière première de l'informatique. Ce sont les données qui remplissent les mémoires, que les réseaux transportent et que traitent les processeurs. La donnée résulte d'une *observation* (ou *mesure*), elle-même conditionnée par une *définition* découlant d'un *choix*.

On est ici tout près de Saussure : alors que le « signe » est un doublet formé d'un concept et d'un son, la « donnée » est un doublet formé d'une définition et d'une mesure. La *sémantique* de la donnée réside dans sa définition ; sa *physique* réside dans la mesure que fournit l'observation et que l'informatique enregistre.

La sémantique de la donnée est tout entière du côté du professionnel qui va se faire assister par l'informatique, car lui seul peut définir ses priorités et indiquer les observations qui sont nécessaires. La qualité de cette sémantique conditionne celle de l'informatique ; c'est dans la maîtrise de cette qualité, dans la construction du langage de l'entreprise, que réside le problème fondamental pour l'informatisation.

Si la sémantique de l'entreprise est mal bâtie, si l'on y code n'importe comment, l'informatisation sera ratée quelle que soit la qualité formelle des programmes informatiques. Or dans beaucoup d'entreprises les identifiants sont de mauvaise qualité, les référentiels mal conçus, les données de référence mal gérées : la plupart des défauts des systèmes d'information ont leur origine non dans l'informatique elle-même, mais dans le manque de rigueur sémantique des maîtrises d'ouvrage.

Ces défauts-là risquent d'échapper à l'attention des informaticiens. Ayant réservé le mot « sémantique » aux composantes abstraite, opérationnelle, dénotationnelle et axiomatique de la « sémantique formelle », il ne leur reste plus pour désigner la sémantique de l'entreprise que l'expression « sémantique informelle », expression entachée de connotations péjoratives comme « floue », « mal définie », « peu rigoureuse » etc. Cela détourne l'entreprise de l'effort nécessaire pour construire son référentiel.

La cloison ainsi érigée entre l'informatique et la sémantique de l'entreprise interdit en pratique de penser l'automatisation, d'articuler convenablement l'automate aux processus de production. Ainsi on ne saura ni négocier, ni même définir la perte en information (au sens de Shannon, voir

page 36) qui est nécessaire pour obtenir un gain en signification, alors que cette démarche constitue le cœur même de la statistique (Volle [211]).

À l'extrême l'informaticien considère les données comme un minerai dont seule importe la volumétrie. Toutes sont traitées pêle-mêle, de la même façon, qu'il s'agisse d'identifiants ou d'attributs, de données saisies, de données intermédiaires ou de résultats - alors que pour la sémantique de l'entreprise les identifiants doivent être parfaits, les données saisies doivent être vérifiées. Si les informaticiens ont explicité pour les échanges de données des critères de qualité formelle comme les propriétés ACID (atomicité, cohérence, isolation, durabilité), rares sont les entreprises qui ont mis en place une administration des données (voir page 265) qui garantisse la qualité de leur sémantique.

La négligence envers la sémantique de l'entreprise est d'autant plus dommageable que l'assistance par l'automate s'est étendue à tous les processus de production et s'articule désormais de façon intime avec le travail des êtres humains. Le succès de l'informatique la confronte à un défi intellectuel : maîtriser la sémantique de l'entreprise, définir judicieusement l'articulation entre l'automate et l'être humain.

* *

L'informatique, grande aventure intellectuelle de notre époque, a attiré quelques-uns des esprits les plus fins et les mieux équilibrés. Ceux-là ne sont nullement gênés par un vocabulaire mal conçu : ils savent ce que les mots désignent et ne sont pas dupes des connotations. Ils sont par ailleurs assez ouverts, curieux et généreux pour s'intéresser à l'entreprise et définir intelligemment sa sémantique. Il existe aussi parmi les maîtrises d'ouvrage, plus rarement il est vrai, des personnes qui savent ce que l'on peut attendre de l'automate et comprennent ses contraintes.

Ces diverses personnes savent éviter les pièges et n'ont donc pas besoin de garde-fou. Mais quand on pense à l'organisation de l'entreprise, à son langage, il faut considérer non pas seulement ces personnes à l'intelligence déliée, mais aussi l'individu moyen dont la compréhension est embrouillée par les faux amis, et encore les dirigeants qui ne sont pas des experts mais doivent pourtant pouvoir prendre des décisions justes. Il faut considérer enfin les personnes intelligentes elles-mêmes, à qui l'on doit épargner le temps et l'effort nécessaires, aussi intelligentes soient-elles, pour se repérer dans un univers confus.

On doit également - car dans une entreprise on ne vit pas parmi les anges du Paradis - ôter aux pervers les armes que leur procure la confusion du vocabulaire et des idées, ainsi que les espaces trop propices que cette confusion procure aux jeux de pouvoir.

7.2 Ingénierie du système d'information

« À analyser les textes les plus anciens (...), avant même que ne s'épanouisse une littérature religieuse ou profane, on peut approcher le pourquoi de l'écriture. Il s'agit généralement d'im-

poser sa marque de propriété, de tenir des comptes, d'établir des listes d'individus, d'animaux ou de biens, de cataloguer en quelque sorte. (...) Ce sont donc des raisons très pragmatiques et concrètes qui ont poussé les hommes à inventer un mode de transmission différent du langage, et non une réflexion théorique. (...) Pourtant, le seul fait d'écrire (...) a profondément modifié la manière de penser de ceux qui ont utilisé ce nouvel instrument. (...) Pouvoir dresser une liste qui n'a plus besoin d'être mémorisée entraîne toute une série de changements dans la manière de raisonner, ainsi que dans la transmission des connaissances. » (Christiane Zivie-Coche, *in* L. Bonfante et coll., *La naissance des écritures* [18], p. 13)

Un système d'information a pour but d'assister les agents de l'entreprise dans leur travail en leur permettant de mobiliser, à travers le réseau, les ressources de puissance et de mémoire que fournit l'automate. Il enregistre le langage de l'entreprise qu'il équipe en outils de classement, recherche, traitement et communication.

Dans tout système d'information diverses logiques fonctionnent conjointement. On peut donc utiliser un modèle en couches pour faire apparaître leur juxtaposition et leur articulation. La délimitation des couches peut toujours se discuter et d'ailleurs on peut représenter un même organisme selon divers modèles en couches selon le point de vue que l'on adopte.

À la base du système d'information se trouve le socle sémantique contenant les référentiels (modèles, nomenclatures, identifiants). Ce socle est géré par l'administration des données. Il fournit à l'entreprise son vocabulaire ; il documente la façon dont elle classe et décrit ses processus, ses clients, ses produits etc. ainsi que les attributs qui les caractérisent.

La plate-forme technique comporte les matériels (machines, réseaux), les dispositifs de commande (système d'exploitation, langages de programmation, progiciels) et les solutions d'architecture (middleware, interfaces etc.).

Le contenu applicatif des systèmes d'information modernes se définit essentiellement par les *objets* ou *composants*, structures qui représentent chacune un être du monde de la nature et articulent les liens vers des enregistrements contenus dans les bases de données, et par les *processus* qui balisent la succession des tâches exécutées sur les composants. Les modèles décrivant les processus et les composants figurent dans le référentiel de l'entreprise. Construire un système d'information, c'est d'abord *modéliser* les composants et les processus.

Le système d'information fournit aux utilisateurs, outre les outils applicatifs, des ressources de bureautique et d'informatique communicante qui leur permettent de produire et communiquer des textes en langage naturel : messagerie, documentation électronique, rédaction coopérative, diffusion sélective, forums etc. L'ensemble de ces outils a été désigné par le terme « groupware », de plus en plus souvent remplacé par « Intranet » (ou « Extranet » si la communication s'étend à plusieurs entreprises).

L'informatique de communication permet d'associer aux données structurées le commentaire qui les rendra intelligibles. Elle s'articule aux ou-

Applications : Une application se définit d'une part par des structures de données, d'autre part par des algorithmes de calcul ^a. Les utilisateurs appellent chaque application par un nom propre (« Claudine », « Chéops », « Galilée » etc.), ce qui lui confère une sorte de personnalité. Dans l'urbanisme du système d'information les applications sont comme des immeubles familiers, parfois imposants mais construits à des dates différentes et entre lesquels il est parfois difficile de communiquer.

Composants : Dans le vocabulaire des langages à objets, un composant est un ensemble de classes articulées autour d'une classe maître ; dans le langage courant, un composant est dans le système d'information la représentation d'un dossier (dossier client, dossier produit, dossier commande etc.).

Passer de l'« application » au « composant », c'est changer de priorité et de point de vue. Avec l'application, la priorité appartient aux traitements, aux algorithmes qui accaparent l'attention du programmeur. Avec le composant, la priorité revient aux données, ou plutôt selon le vocabulaire des langages à objets aux « attributs » observés sur chaque dossier ; le programme est organisé de sorte que la cohérence des attributs d'un même objet soit préservée contre toute intervention intempestive ^b.

Processus : Le « processus » est la succession des tâches qui contribuent à une même production. Alors que les applications ne contenaient pas d'information sur les processus, les systèmes d'information automatisent aujourd'hui le parcours du dossier entre les divers agents qui doivent le traiter (traitement d'une commande, d'une demande de crédit, d'une lettre de réclamation etc.).

^a Cf. le titre de l'ouvrage de Niklaus Wirth [219], créateur du langage Pascal : *Algorithms + Data Structures = Programs*

^b C'est ce que l'on appelle « encapsulation ».

tils applicatifs : l'aide contextuelle utilise la documentation électronique, les « workflows » équipent les processus en utilisant les mécanismes de la messagerie.

Les qualités essentielles que doit posséder un système d'information sont :

- la *pertinence* (adéquation aux besoins des utilisateurs),
- la *sobriété* (il est inutile et coûteux de mettre en service des fonctionnalités qui resteront inutilisées),
- la *cohérence* (sans cohérence, on ne peut pas parler de système!).

Nous avons mentionné la cohérence en dernier parce qu'il s'agit d'une qualité *formelle*, nécessaire mais non suffisante et plus facile à contrôler et administrer que les deux autres.

Pour obtenir un système d'information possédant ces qualités il faut avant toute réalisation procéder à des consultations et expertises, modéliser les processus et composants, spécifier les *fonctionnalités* que le système d'information doit fournir.

Le poste de travail (ordinateur personnel en réseau) fournit à l'utilisateur l'interface (écran, clavier) à travers laquelle il accède au système d'information.

Le but de l'entreprise étant d'obtenir un couple « homme - machine » efficace dans l'optique du « travail assisté par ordinateur », il convient de définir le poste de travail de telle sorte qu'il équipe convenablement l'utilisateur et de former celui-ci à son maniement. Il ne faut pas entendre ici le mot utilisateur au singulier : si l'on considère un processus, l'« utilisateur » désigne l'ensemble des personnes qui sont organisées pour faire fonctionner le processus. L'utilisateur, c'est donc l'« être humain organisé » en vue de la production. Souvent, la conception du système d'information et sa mise en œuvre impliquent une redéfinition de l'organisation, c'est-à-dire des missions attribuées aux entités de l'entreprise et aux personnes ; la mise en œuvre de cette réorganisation s'appelle « conduite du changement ».

* *

L'ingénierie des systèmes d'information est un cas particulier de l'ingénierie de système (Meinadier [130]). On y retrouve les mêmes notions (maîtrise d'ouvrage et maîtrise d'œuvre, conduite de projet, spécification des exigences etc.). Cependant l'ingénierie de système est plus difficile dans le système d'information que dans les projets techniques, fussent-ils très compliqués.

Un projet technique comme la conception d'un avion ou d'une centrale nucléaire est en effet réalisé par des ingénieurs qui expriment leurs exigences en termes observables et mesurables, et qui sont soucieux de traiter les questions de *physique* que pose la conception et la fabrication du *produit* de telle sorte que celui-ci réponde à ces exigences.

Par contre un projet de système d'information de gestion concerne non un produit, mais le fonctionnement d'une *organisation*. Il n'est pas facile de définir les critères qui permettraient d'évaluer sa réussite, et il existe un écart entre l'organisation humaine, dont le flou est à la fois naturel et entretenu, et le logiciel dont le fonctionnement est automatique :

- un spécialiste de l'ingénierie de systèmes, invité à mettre en place un système d'information, demanda que l'on indiquât d'abord la répartition des pouvoirs de décision. Son intervention s'est arrêtée là, les dirigeants préférant laisser implicite la répartition des responsabilités ;

- le responsable de la maîtrise d'ouvrage d'une grande entreprise m'a dit qu'il devait consacrer 95 % de son temps de travail à des questions de pouvoir, et qu'il ne lui en restait que 5 % pour traiter les questions relatives au processus de production.

Ainsi, alors que l'ingénierie de système donne une place importante à l'identification des exigences, au suivi de leur réalisation tout au long du projet et enfin à leur vérification lors de la recette finale, on le fait rarement pour un système d'information. On devrait pourtant à tout le moins imposer à la maîtrise d'œuvre de formaliser les spécifications sous forme d'exigences, avec les méthodes de vérification associées, et de produire les tableaux de bord qui permettront d'en assurer le suivi.

* *

Comme le système d'information s'insère dans l'organisation et touche à la répartition des pouvoirs légitimes, il faut l'ancrer sur des choses aussi stables que possible. Il est salubre de prendre pour point de départ les *produits* que l'entreprise élabore, puis leurs *processus de production*, ensuite les *livrables* (ou *produits intermédiaires*) que fournissent des sous-processus, enfin les *activités* qui s'enchaînent dans le parcours de chaque processus.

La démarche se fonde ainsi sur la *physique* de l'entreprise - processus de production et de commercialisation, relations avec les clients et partenaires - et non sur la répartition des pouvoirs que découpe l'organigramme et que l'on nomme, par abus de langage, *organisation*. Les applications informatiques deviennent alors invariantes par rapport à cette « organisation » qui, elle, est de plus en plus évolutive : un changement de l'« organisation » entraînera une simple modification de l'affectation des activités aux diverses entités de l'entreprise.

Les exigences s'expriment en termes de qualité des produits, de qualité des processus (maîtrise des délais, satisfaction des clients) et d'efficacité (consommation des ressources), toutes choses qu'il est possible d'observer et de mesurer. Il faut donc non seulement que le système d'information outille le processus en automatismes, mais aussi qu'il produise les indicateurs qui, rendant sa qualité visible par tous, inciteront à la maintenir.

L'ingénierie doit, dans les systèmes techniques, satisfaire toutes les exigences initiales (amendées par d'éventuelles demandes de dérogation formellement acceptées par le maître d'ouvrage) : ces exigences, exprimées par des ingénieurs, résultent d'une sélection sévère. Dans les systèmes d'information, par contre, les exigences initiales sont souvent démesurées. Il faudra savoir ne retenir parmi elles que les 20 % vraiment indispensables, leur sélection devant être dûment justifiée.

Il faut en ingénierie de système se défier des connotations du mot « optimiser » et du mot « rationnel » : mieux vaut s'efforcer d'être *raisonnable*, terme qui a d'ailleurs lui aussi pour racine le mot *raison*.

Il est en effet impossible, lorsqu'il faut satisfaire quelques milliers d'exigences, d'introduire l'ensemble des contraintes dans une équation qui fournirait le paramètre économique qu'il s'agit de maximiser. Peut-on d'ailleurs définir le *meilleur* avion, la *meilleure* automobile? Dans un tel contexte, « optimiser » ne peut pas vouloir dire que l'on cherche à maximiser une fonction objectif: il s'agit plutôt de faire une analyse de la valeur sur les choix successifs selon une approche qui considère l'ensemble du cycle de vie du produit.

L'ingénierie de système implique donc que l'on s'attache à raisonner sur des choix et à les justifier en tenant compte des points de vue des acteurs concernés. Beaucoup de systèmes d'information sont devenus des monstres inexploitable parce que l'on ne s'était pas interrogé sur la pertinence de la demande des utilisateurs.

Il est *raisonnable* de chercher à faire le travail le moins lourd possible tout en satisfaisant convenablement les besoins. Cela implique de ne jamais tenter d'automatiser des tâches que l'être humain accomplit mieux que l'ordinateur. La FAA (*Federal Aviation Administration* des États-Unis), qui a tenté d'automatiser le travail des contrôleurs aériens, a dû arrêter ce projet alors qu'il avait coûté plusieurs centaines de millions de dollars. Il ne faut pas non plus tenter d'automatiser les arbitrages politiques, les décisions stratégiques auxquelles le système d'information ne peut fournir que des tableaux de bord et des simulations. Par contre, on peut automatiser efficacement une *aide* à la décision *opérationnelle* (quart-opération du transport aérien, décision de prêt dans une banque, cellule de crise).

Bien souvent, on ne peut pas assigner de limite rationnelle à la richesse d'un outil, au détail d'un référentiel etc. Dans ce cas, il sera raisonnable de borner les exigences en se fixant une limite arbitraire en budget et en délai. On pourra par la suite enrichir l'outil si le besoin s'en fait fortement sentir.

* *

Le système d'information est le *langage* de l'entreprise, un langage articulé à son action. Il est organiquement lié à son positionnement, à ses priorités: il exprime sa *personnalité*. L'entreprise le secrète tout comme une civilisation secrète sa langue. Il n'est donc pas étonnant que sa définition révèle des enjeux, suscite des conflits, s'accompagne de malentendus. Les compétences de l'ingénieur, la rigueur avec laquelle il applique les principes de l'ingénierie, ne suffisent pas pour concevoir un système d'information: il faut aussi qu'il possède une sensibilité de sociologue et des compétences en linguistique. Il faudra qu'il sache « manipuler pour la bonne cause », dans le droit fil de l'École de Palo-Alto (Marc et Picard [160]). Ceci est vrai d'ailleurs non seulement pour les systèmes d'information, mais aussi pour ceux des projets techniques et scientifiques qui impliquent plusieurs entreprises, plusieurs pays, plusieurs cultures.

La conception d'un système d'information suppose une spécification précise, sans quoi on abandonne au programmeur le choix de ce qui devra être implémenté. Les choix fondamentaux, qui relèvent de l'analyse de la valeur, sont *in fine* du ressort de la maîtrise d'ouvrage qui seule peut évaluer la rentabilité d'un projet. L'apport majeur de l'ingénierie de système réside

dans la clarté de la justification des choix du maître d'ouvrage. Jean-Pierre Meinadier plaide pour une maîtrise d'ouvrage compétente et rigoureuse. L'expérience lui a montré que la maturité de la maîtrise d'ouvrage était le facteur décisif de la réussite d'un projet, c'est-à-dire de la satisfaction des utilisateurs et des autres parties prenantes.

7.3 Approche linguistique

Les Grecs distinguaient dans la réalité observable la φύσις et la θέσις, la « nature » et la « thèse ». La φύσις, indépendante de la volonté collective des hommes, relève de l'ordre régulier du monde : c'est le *monde de la nature* ; la θέσις recouvre ce qui dépend de la volonté collective, que celle-ci soit ou non consciente et explicite : c'est le *monde de la pensée* au sens le plus large.

Lorsqu'on considère une entreprise, on doit de même distinguer les « événements du monde de la nature », dont le constat ou l'anticipation déclenchent son activité (réception d'une commande ou d'une réclamation, innovation technique, initiative d'un concurrent etc.) et les processus internes selon lesquels elle élabore sa réponse à ces événements. Cette réponse consiste elle-même en la production de nouveaux événements, qui *agissent* sur le monde de la nature.

L'examen des processus conduit à distinguer dans l'entreprise d'une part la couche « physique » où résident la fonction de production et les facteurs de production (machines, personnel, matières premières) ainsi que la relation avec les clients et les fournisseurs ; d'autre part la couche « organisation » où se définissent les entités légitimes et se délimitent les pouvoirs de décision (investir, diversifier les produits, former, promouvoir ou sanctionner les personnes etc.).

Le système d'information équipe les processus auxquels il apporte les ressources de l'automate. Il est lui aussi tout à la fois :

- φύσις : support physique de la circulation des idées au sein de l'entreprise (de même que la modulation de l'onde porteuse par le signal sonore est, sur un réseau téléphonique, le support physique de la conversation entre des personnes),
- θέσις : cadre conceptuel et théorique *a priori* qui, fondant le discernement des agents et leur capacité de décision, leur permet de percevoir les événements du monde réel et d'agir sur lui.

Le système d'information apparaît alors comme un *système de signes* (Saussure [179]), comme le *langage de l'entreprise*, un langage entièrement orienté vers l'action. La succession des langages informatiques qui s'empilent du microcode aux applications⁴ culmine dans la définition conceptuelle et

4. Les couches du langage qui s'empilent jusqu'au microcode visent à assurer (1) la traduction progressive des ordres donnés à la « machine virtuelle » en instructions élémentaires que l'automate sera capable de traiter ; (2) l'évolutivité et la maintenance du logiciel grâce à la clarté du découpage en modules, à la gestion de configuration, à la documentation, au nommage etc., évolutivité qui doit permettre de répondre au moindre coût aux chocs provenant (a) de l'environnement réglementaire (droit, fiscalité) ; (b) de

fonctionnelle du système d'information⁵. Le système d'information offre à ses utilisateurs une machine virtuelle où les concepts proches de l'action sont explicités, disponibles et manipulables ; il structure l'assistance qu'apporte l'automate programmable à l'opérateur.

* *

La physique de l'entreprise suppose l'adaptation à un monde en évolution : les techniques changent ainsi que la réglementation, les concurrents prennent des initiatives, les besoins des clients évoluent. L'ingénieur soucieux d'efficacité souhaite donc que le langage de l'entreprise soit aussi souple que le volant d'une automobile et que l'organisation évolue sans retard. Cependant l'organisation est en place, les missions sont définies, leurs responsables désignés : l'entreprise attend que l'ingénieur agisse dans ce cadre et selon ce qu'il autorise.

Chacune de ces deux exigences est rationnelle. Une entreprise rigide, indifférente aux évolutions du monde réel, deviendrait à la longue inefficace, mais par ailleurs une entreprise dont l'organisation serait modifiée sans cesse ne pourrait pas stabiliser son langage et cela déconcerterait ses agents : beaucoup d'entre eux partiraient, les compétences ne pourraient pas s'accumuler. Un chantier permanent ne peut pas être efficace.

La solution raisonnable ne peut résulter que d'un arbitrage entre les exigences de la physique et celles de l'organisation. La qualité d'un dirigeant s'évalue selon son aptitude à assurer cet arbitrage : le bon dirigeant est à la fois attentif à l'organisation et vigilant envers les techniques et le marché.

Dans certaines entreprises toutefois, les dirigeants vivent dans un monde qui relève d'une sociologie spécifique et les sépare de la physique de l'entreprise⁶. Le système d'information est alors non pas une articulation entre l'organisation et la physique, mais l'enjeu d'une lutte entre elles.

Le *discours* d'une entreprise portera toujours exclusivement sur sa physique : objectifs d'efficacité, de compétitivité, de création de valeur etc. ; mais il sera souvent plaqué sur une réalité toute différente, que le système d'information révèle comme le ferait une radiographie. Quand les décisions des dirigeants sont déterminées par la seule organisation, les forces qui concourent à l'entropie du système d'information, à l'éclatement du langage, jouent sans contrepoids. Le référentiel s'éparpille en de multiples tables de codage spécifiques chacune à un domaine ; tout codage se diversifie encore en dialectes locaux, chaque région l'interprétant à sa façon ; certaines données seront mal codées, les agents opérationnels jugeant leur qualité indifférente ; les lacunes dans la réalisation des applications seront rattrapées sur le terrain

l'environnement économique (évolution de la clientèle, des consommations intermédiaires, des techniques, des fournisseurs, des partenaires, des concurrents).

5. Pour pouvoir se comprendre et échanger leurs intentions les êtres humains enrichissent le langage conceptuel du système d'information par le symbolisme du langage connoté, auquel l'informatique de communication fournit un vecteur puissant.

6. Une situation analogue se retrouve à d'autres niveaux : les personnes qui détiennent le pouvoir politique peuvent être attentives à la société, vigilantes envers le monde extérieur ; mais elles peuvent aussi relever d'une sociologie spécifique qui les distingue et les isole.

par des ressaisies et traitements manuels pénibles etc. La plate-forme informatique, elle aussi négligée, connaîtra les pannes sans responsable identifié, les « buffers » qui débordent et l'insécurité.

Lorsqu'on veut agir pour faire évoluer le système d'information d'une entreprise, on rencontre deux obstacles : l'un réside dans la compréhension des enjeux physiques de l'entreprise, de ses priorités, et de la configuration qui doit en résulter ; l'autre réside dans l'organisation, qui résiste à la communication entre les personnes de l'entreprise ou s'oppose à des changements qu'elle juge défavorable pour ses intérêts.

7.3.1 Physique de l'entreprise

Le fonctionnement de l'entreprise appartient, bien que celle-ci soit une création historique, au règne de la nature. Il obéit à des lois contraignantes : celles du marché sur lequel l'entreprise présente son offre, celles du crédit qui finance son activité, celles de l'organisation interne de la production.

Notre connaissance de ces lois est limitée, comme dans tous les autres domaines de la nature. Nous pouvons les approcher par l'expérience, les explorer par le raisonnement, nous armer de règles et de principes, cela n'exclura jamais les surprises et nous devons rester vigilants : même si le raisonnement fournit des *poignées* qui permettent d'agir sur la nature la complexité de celle-ci le dépasse.

Le système d'information concourt à la vigilance en même temps qu'il permet de maîtriser le parcours des processus de production. Sa conception doit elle aussi obéir à des principes ; certains d'entre eux visent à éviter des incohérences, et on ne penserait pas même à les mentionner si, par défaut d'intuition envers la physique de l'information, les erreurs n'étaient pas si courantes.

Mentionnons ces principes élémentaires pour n'y plus revenir : définir les domaines d'action, les processus de production de valeur ajoutée, les « populations » d'entités concernées par ces processus, les « classes d'objets » à utiliser pour décrire ces populations ; organiser les processus de façon à éviter les doubles saisies, doubles identifications et connexions répétées à des applications diverses ; éliminer les synonymes et les homonymes ; construire les référentiels (identifiants, définitions des données), et gérer les données de référence, de sorte que la sémantique du système d'information soit maîtrisée...

Quelle armée mettrait en campagne des soldats qui ne sauraient ni marcher, ni faire usage de leurs armes, ni distinguer l'ami de l'ennemi, et des officiers qui n'auraient reçu aucune formation à la tactique ? Une entreprise dont le système d'information viole ces principes élémentaires ressemble à une telle armée. Pourtant l'expérience montre que tout « élémentaires » que soient ces principes, ils sont rarement respectés.

Lorsqu'on a su éviter les incohérences, il reste à définir le système d'information qui correspond à la stratégie de l'entreprise. Ici les choses deviennent subtiles et l'intuition est souvent déroutée. Supposons par exemple que votre entreprise exploite une intermédiation sur un marché. Vous avez à gérer deux

relations : avec les offreurs, avec les demandeurs ; le cœur de votre activité, c'est de former des couples offre - demande de bonne qualité, c'est-à-dire qui suscitent une transaction satisfaisante pour les deux parties.

Supposons maintenant qu'à l'offre principale s'ajoute une offre secondaire. C'est le cas si l'on considère le marché du travail, sur lequel l'offre principale est l'offre d'emploi qui répond à la demande d'emploi, et l'offre secondaire est l'offre de formation professionnelle : on peut en effet, si l'on ne trouve pas immédiatement d'offre pour un demandeur, chercher pour celui-ci la formation qui lui permettrait d'être en meilleure posture sur le marché du travail.

Alors l'intermédiation doit mettre en relation non plus deux populations (offres et demandes d'emploi) mais trois (offres et demandes d'emploi, offres de formation) ; on doit traiter non plus *une* relation entre deux populations, mais *trois* relations entre trois populations. L'ajout d'un pôle relationnel, qui semblait n'accroître la complexité du système d'information que de 50 % (passer de deux à trois pôles), l'accroît en réalité de 200 % (passer d'une à trois relations) (figure 7.1). Pour anticiper l'effort que demande une telle évolution, il faut avoir une intuition immédiate et exacte de la physique de l'entreprise.



FIG. 7.1 – Passage de deux à trois pôles

L'implication du système d'information dans la relation avec le client fait elle aussi l'objet de choix stratégiques, d'ailleurs révélateurs des priorités de l'entreprise. Si une banque identifie non le client lui-même mais les divers comptes qu'elle lui ouvre, de sorte qu'il lui sera pratiquement impossible de faire le tour de l'ensemble de ses relations avec un client particulier, elle révèle que sa priorité n'est pas de connaître ni de comprendre le client mais de gérer séparément les divers outils commerciaux qu'elle utilise. Son regard est dirigé non vers le client, quoi qu'elle puisse dire, mais vers sa propre organisation : l'organisation interne, les plates-bandes des diverses directions, voilà ce qui importe. Il en sera de même de l'opérateur télécoms qui identifie non ses clients, mais des lignes etc.

Si une banque décide de confier à la même direction la responsabilité de la conception des services offerts à la clientèle et celle du *back office* correspondant, elle donne à cette direction la possibilité de vérifier la façon dont ses prescriptions sont appliquées et les effets qui en résultent (puisque ce qui se passe dans la première ligne a des conséquences visibles dans le back-office), ce qui lui permettra de réagir rapidement et d'accélérer la maturation de son offre de services.

Si une entreprise décide de gérer son offre sur l'Internet indépendamment de son offre traditionnelle (qui passe par des commerciaux, par des agences sur le terrain) et ainsi d'éclater la relation avec le client selon le vecteur que celui-ci emprunte, cela veut dire qu'elle souhaite en réalité créer une nouvelle entreprise dédiée au service sur l'Internet et la rendre indépendante : car une entreprise ne peut pas impunément présenter au client deux personnalités différentes et mutuellement étanches.

* *

La physique de l'entreprise se décalque dans la physique de l'information, et réciproquement les choix faits pour le système d'information, s'ils sont exempts d'incohérence (c'est-à-dire s'ils respectent les principes élémentaires), révèlent des choix stratégiques implicites. Si, en bonne logique, la définition de la stratégie précède celle du système d'information, en pratique les deux démarches s'appuient mutuellement.

« La guerre, dit Clausewitz [37], est la continuation de la politique par d'autres moyens ». De même le système d'information est la poursuite de la stratégie de l'entreprise par d'autres moyens. Il arrive que la politique, notamment la géopolitique, soit déterminée par l'équilibre des forces militaires en présence. Il peut aussi se faire que, par un retournement de l'ordre des priorités, la stratégie soit déterminée par les possibilités qu'offre le système d'information : dans le transport aérien, les systèmes informatiques de réservation ont bouleversé les stratégies.

L'orientation stratégique de l'entreprise est, comme celle d'une nation, affaire d'adaptation aux circonstances ; elle doit épouser la *propension des choses* (Jullien [93]). On ne saurait l'évaluer selon les seuls critères de la psychologie en accusant celui-ci de timidité, celui-là de témérité etc. : on est là dans le domaine des choses naturelles, complexes, dont la connaissance est difficile à communiquer, dont la maîtrise (toujours relative) s'acquiert par un talent spécial conjugué à l'expérience. La vigilance, le sens du possible, supposent une attitude fondamentalement modeste : le stratège doit être prêt à apprendre, à tirer les leçons de l'expérience, à mouler sa décision sur la situation présente et prévisible.

7.3.2 Articulation entre physique et sociologie

Nota Bene : Par « sociologie de l'entreprise » nous entendons ici non l'étude sociologique de l'entreprise, mais la détermination sociologique du comportement des acteurs (voir page 543).

* *

L'entreprise sera d'autant moins « naturelle », d'autant plus « sociologique », qu'elle accordera plus d'importance à l'organisation et moins d'importance à la physique ; que son regard sera davantage tourné vers l'intérieur, et non orienté vers le client ; que les décisions seront dictées par le souci d'éviter les « problèmes de personnes » plus que par la recherche de l'efficacité.

Nous avons déjà évoqué les indices qui, dans le système d'information, révèlent la véritable priorité de l'entreprise. Mais il existe d'autres indices plus faciles à observer. Ainsi toute entreprise qui proclame son intention de « mettre le client au cœur de l'entreprise » révèle qu'elle regarde son propre nombril : si elle accordait au client l'importance qu'il mérite, cela lui serait tellement naturel qu'elle n'aurait pas besoin de le dire.

La sociologie de l'entreprise est portée par ses forces dirigeantes - il s'agit bien sûr des dirigeants, mais aussi des réseaux (syndicats, corporations, partis politiques) qui ont pu s'emparer d'une part du pouvoir, comme c'est souvent le cas dans les entreprises anciennes et de grande taille. Cette sociologie interne est soumise à des chocs provenant d'autres sociologies, extérieures à l'entreprise et qui la font vibrer comme une caisse de résonance : celles des actionnaires, des clients, des pouvoirs politiques et réglementaires, de l'opinion etc.

* *

Traiter efficacement la physique de l'entreprise aura inévitablement des conséquences sociologiques. Vous introduisez l'e-business? Cela modifie le partage des pouvoirs, le rôle des directeurs régionaux, les missions des commerciaux etc. Vous mettez la documentation de l'entreprise sur l'Intranet? Cela ôte à certaines personnes un monopole qui garantissait la pérennité de leur emploi, cela modifie la balance du savoir entre la direction générale et les directions régionales. Vous organisez la maîtrise d'ouvrage des systèmes d'information? Cela crée des inquiétudes du côté de l'informatique. Peu importe que les craintes soient justifiées ou non : elles existent et suffisent pour susciter des réactions défensives.

Il arrive que les réalisations les plus évidentes, les plus nécessaires du point de vue de la physique de l'entreprise, soient entravées, ajournées, repoussées, d'une façon qui semble absurde à ceux qui ne discernent pas les ressorts sociologiques à l'œuvre. On entend alors souvent un ingénieur dire : « je me demande si je suis fou, ou si ce sont les autres qui le sont », phrase caractéristique.

Il ne faut pas être naïf devant la sociologie de l'entreprise : ceux qui sont surpris ou scandalisés par les réactions de défense des personnes montrent qu'ils n'ont pas compris comment une entreprise fonctionnait sur le plan symbolique, comment les angoisses, les ambitions, les rêves de chacun y prennent forme et déterminent les comportements.

Souvent, la sociologie de l'entreprise est imposée au responsable du système d'information avec une force écrasante, par la voie hiérarchique et sous forme d'injonctions, de rappels au bon sens, à l'évidence et au « pragmatisme ». Il ne peut certes pas se permettre de la négliger. Quelle doit être alors son attitude?

Il convient certes d'être intellectuellement *modeste* envers la physique de l'entreprise, dont la complexité nous dépasse. Mais il ne faut pas être *timide* envers la sociologie de l'entreprise, même si celle-ci est portée par les forces qui dirigent (qu'il s'agisse des dirigeants ou des réseaux). Elle existe ; dans la mesure où elle détermine les comportements, elle est l'une des composantes

de la physique de l'entreprise : il faut en tenir compte comme l'on tient compte de ses autres composantes. Mais elle ne saurait déterminer ni les principes, ni les règles, car elle leur est indifférente.

Chaque fois que la sociologie entre en conflit avec les règles élémentaires dont le respect conditionne la cohérence du système d'information (lorsque les disputes entre services s'opposent à la mise en place d'une administration des données etc.), *il faut la combattre* résolument : si on la laisse faire, l'entreprise n'aura pas de système d'information et sera privée d'un organe essentiel à son existence.

Chaque fois que la sociologie usurpe la place de la stratégie (que le « pas de vagues » a le pas sur l'efficacité), *il faut la combattre* parce que sinon l'entreprise risque d'être paralysée comme si elle avait la maladie de l'homme de pierre (myosite ossifiante progressive, caractérisée par l'ossification du tissu musculaire).

Chaque fois que la sociologie pousse l'entreprise à orienter son regard vers son nombril, à faire passer l'organisation interne avant la considération du client ou, s'il s'agit d'une administration, à faire prévaloir la conception corporatiste du service public sur le service du public, *il faut lutter* pour remettre les priorités sur pied.

Cela demande de l'intrépidité. Cela va au rebours des consignes (« ne pas chercher à comprendre », « s'écraser », « ne pas faire de vagues ») que l'on dispense si libéralement dans l'entreprise.

Les difficultés que recèle la physique de l'entreprise sont souvent mal perçues, ou peu comprises, même quand l'entreprise dépense chaque année des milliards d'euros pour son système d'information ; par contre les contraintes qu'impose la sociologie de l'entreprise jouent à chaque instant et leur poids est parfaitement perceptible. C'est donc envers ces contraintes-là que l'on vous invitera à la modestie, au réalisme, à la patience, au prétendu pragmatisme, comme s'il s'agissait de choses « dures », contraignantes, de la φύσις, alors qu'il ne s'agit que de comportement et de décision et que cela relève donc de la volonté collective, de la θέσις.

On vous demandera au besoin de sacrifier vos exigences de rigueur, de sacrifier les principes élémentaires (et donc, en violant la cohérence, de violer la nature elle-même), pour être plus accommodant, plus « souple » envers la sociologie. Or la nature se venge toujours quand on la viole.

* *

Il ne faut pas prendre la sociologie de front, ce serait suicidaire : en ce sens il faut en effet savoir être « souple ». Mais du point de vue des orientations, des priorités, les choses doivent être très claires : *modestie* devant la physique ; *intrépidité*, refus de l'intimidation, devant la sociologie.

Le cérémonial qui entoure les dirigeants, les signes vestimentaires et physiques de leur importance (âge, corpulence, lunettes, habillement, décorations, coiffure, voiture), le luxe des bureaux et salles de réunion, le sérieux des huissiers et assistantes, la qualité du vocabulaire, de l'élocution, de la tenue à table, tout cela est fait pour intimider et risque d'inhiber l'expression des nécessités physiques de l'entreprise au bénéfice de sa sociologie.

Mais cet obstacle symbolique est superficiel. Soyez patients mais intrépides. Soyez assez pédagogues pour faire « réaliser » par les dirigeants les contraintes, les complexités de la physique de l'information, de la physique de l'entreprise. Sachez faire émerger ces questions à l'horizon de leurs préoccupations stratégiques. À terme, et s'ils vous laissent survivre, ils vous sauront gré de les avoir incités à relativiser les contraintes de la sociologie.

7.4 Administration des données

On appelle « administration des données » la fonction de ceux qui, dans une entreprise, sont chargés de veiller à la qualité *sémantique* du système d'information : absence de synonymes et d'homonymes ; entretien de la cohérence des codages ; accessibilité et clarté de la documentation etc.

Les spécialistes reconnaissent tous l'importance de l'administration des données pour la qualité du système d'information. Pourtant il est difficile de la mettre en place. Elle nécessite un « travail de bénédictin » assidu et austère et il n'est pas aisé de trouver des volontaires pour le réaliser. Il est d'ailleurs délicat de doser l'effort : comme il n'existe pas de limite logique au détail de la documentation, seul le bon sens permet de s'arrêter lorsqu'on a atteint la précision raisonnable. S'agissant enfin d'une tâche fondamentale pour l'architecture mais dont les effets se font sentir à terme, il est pratiquement impossible d'évaluer sa rentabilité ; elle sera donc souvent classée au second rang des urgences, parmi ces travaux auxquels on pense toujours et que l'on ne réalise jamais.

L'administration des données a des adversaires : ceux qui n'aiment pas « les vagues » s'en méfient car elle risque de poser à l'entreprise des problèmes « politiques » ; ceux qui pensent que tout étant relié avec tout, il ne convient pas d'établir dans la pensée des classifications aux contours nets, et qui se défient d'un travail « intellectuel » qui vise à introduire de l'ordre parmi les concepts. J'ai connu un directeur général qui disait « l'administration des données est un travail intellectuel, *donc* elle est inutile ».

* *

Une *donnée*, c'est le couple logique formé par une *définition* (sémantique) et une *mesure*, la mesure étant caractérisée par le type de la donnée (booléen, entier, réel, qualitatif, textuel etc.), la périodicité, le délai de mise à jour, ainsi que par l'identité de la personne morale⁷ habilitée à mettre la donnée à jour et, si nécessaire, à faire évoluer sa définition (« propriétaire de la donnée »).

La donnée se transforme en *information* lorsqu'elle est communiquée à un être humain capable de l'interpréter, tout comme une gouttelette d'eau en surfusion se transforme en givre au contact d'une surface solide (voir page 36).

7. Entité de l'entreprise (direction, service, mission etc.), par différence avec la « personne physique » qui est un individu.

Les définitions sont contenues dans des *référentiels*. L'administrateur des données est la personne morale garante de la qualité des référentiels et de leur bonne utilisation, ainsi que des informations concernant la mesure et l'identité du propriétaire de chaque donnée.

Cette fonction situe l'administrateur de données tout près des pouvoirs de décision ; en effet, lors de la construction d'une nomenclature et surtout lors de l'identification du propriétaire d'une donnée se définissent les frontières entre les entités de l'organisation. Dans certaines entreprises le rôle de l'administrateur de données est si délicat que l'on a jugé bon de le rattacher au président ou au directeur général.

* *

Par « référentiel » du système d'information d'une entreprise on entend :

- l'ensemble des règles, documents et bases de données concernant les identifiants et nomenclatures⁸ utilisés par le système d'information,
- les règles d'administration du partage des références par les diverses composantes du système d'information.

Expliquons ce que nous entendons par « identifiant » et « nomenclature ». On peut décrire l'entreprise comme un ensemble de « domaines » relatifs chacun à la production d'une ensemble de produits ; les tâches réalisées dans ces domaines s'enchaînent selon des « processus » qui articulent des « activités » réalisées par des êtres humains qu'assistent des automates et qu'outillent des machines.

Tout processus concerne des ensembles (ou « populations », que l'on nomme aussi « entités ») d'êtres que l'on peut appeler « individus » (« instances ») en prenant ce terme dans un sens large : les clients, commandes, factures, produits, personnes de l'entreprise, partenaires et entités de l'organisation sont ainsi des « individus » formant des « populations⁹ ».

Pour construire le système d'information d'une entreprise on part de la définition de ces populations. Le « référentiel » de l'entreprise sera le support de cette définition (quand on dit « le référentiel » tout court, c'est pour désigner l'ensemble des référentiels de l'entreprise ; l'annuaire d'entreprise est un référentiel particulier, le référentiel des personnes). Un référentiel se présente sous deux formes : une forme documentaire (papier ou électronique) pour les utilisateurs humains ; une forme physique (base de données) pour le traitement informatique.

La première indication que donne le référentiel, c'est *la liste des populations* et leur définition. Puis il faut identifier les individus qui composent chaque population. L'identifiant, clé associée à un individu, permet de retrouver les données qui concernent aux diverses étapes de son cycle de vie.

À chaque individu sont associées des données (« attributs ») observées et mises à jour périodiquement ou en continu. À chaque donnée est associée

8. On utilise souvent des synonymes du mot « nomenclature » : « classification », « typologie », « codage », « table », « taxinomie » etc.

9. On pourrait dire, en utilisant le vocabulaire de l'informatique, des « objets » appartenant à des « classes ».

un *type* logique: une donnée peut être quantitative (revenu, poids etc.), qualitative (métier, commune de résidence etc.), qualitative ordinale (classe d'âge d'une personne, tranche d'imposition), textuelle (commentaire); ce peut être une image (photographie de la personne, plan d'un bâtiment), une date, une adresse postale ou électronique, un nom propre etc¹⁰.

La mesure d'une donnée quantitative est un nombre (de type entier, rationnel ou réel). La mesure d'une donnée qualitative est un codage caractérisant l'affectation (classement) d'un individu à une classe d'une nomenclature¹¹

Pour définir la mesure d'une variable quantitative il suffit de préciser son type et, éventuellement, les bornes des valeurs qu'elle peut prendre. Par contre, pour définir la mesure d'une variable qualitative, il faut fournir une *nomenclature*. Une nomenclature est une partition de la population sans omission ni double emploi. À chaque classe de la partition est associé un code. La nomenclature peut comporter plusieurs niveaux (ainsi le code géographique contient les niveaux îlot, commune, canton, département, région) emboîtés les uns dans les autres (une classe d'un niveau donné appartient à une et une seule classe de niveau supérieur).

* *

L'examen des identifiants révèle des priorités *de facto* souvent différentes de celles que l'entreprise prétend ou souhaiterait avoir. Beaucoup d'entreprises attribuent à leurs clients des identifiants qui identifient non le client, mais un équipement qui caractérise le service rendu à celui-ci: ainsi un opérateur télécoms identifie non le client lui-même, mais la ligne téléphonique (identifiée par son numéro, et à laquelle le nom et l'adresse du client seront attachés comme des attributs); une banque identifie non le client, mais le compte (c'est le cas du RIB). Ces conventions rendent difficile le rassemblement des données concernant un même client et ils révèlent la priorité de ces entreprises, qui s'intéressent plus à leur organisation interne qu'au client et à ses besoins. La priorité qu'elles donnent à l'organisation interne, voire à l'organigramme, les empêche de considérer en priorité leur marché, la satisfaction des clients, ou encore l'état de l'art de leur métier et les bonnes pratiques des concurrents.

Il arrive aussi que l'on confonde identifiant et attribut en introduisant dans l'identifiant des éléments porteurs d'information. Si l'on introduit par exemple un élément géographique dans l'identifiant d'un client (numéro du département etc.), on devra changer l'identifiant lorsque le client déménagera, ce qui rendra difficile le suivi historique de son dossier. L'INSEE, avant la mise en place du fichier SIRENE, identifiait les établissements avec un code comportant l'indication de l'activité principale; il fallait changer l'identifiant d'un établissement lorsque son activité principale changeait.

10. Il est possible de transformer une donnée quantitative en donnée qualitative ordinale en attribuant un code à des intervalles de valeur.

11. La « classe » d'une nomenclature n'est pas la même chose que la « classe » d'un langage orienté objet. La première désigne une catégorie de la nomenclature, la seconde désigne l'ensemble d'une population.

Il arrive enfin que l'on réutilise pour un nouvel individu l'identifiant d'un autre individu arrivé en fin de cycle de vie. Ainsi l'ANPE a réutilisé, pour identifier ses agences, les identifiants d'agences supprimées. Cela oblige, lors de l'examen de l'historique des données concernant un individu, à vérifier s'il s'agit continûment du même.

Il importe donc de définir correctement la population dont il s'agit d'identifier les individus : il ne faut pas confondre le client avec le service qui lui est rendu, avec le produit qui lui est fourni ni avec le contrat que l'on a passé avec lui. Il faut aussi construire des identifiants pérennes, qui resteront affectés à l'individu pendant tout son cycle de vie, et s'interdire de réutiliser un identifiant après la fin du cycle de vie. Mieux vaut enfin ne pas confondre le rôle de l'identifiant et le rôle des attributs : l'identifiant ne doit être porteur d'aucune information.

D'un point de vue technique, il résulte de ces règles que la meilleure façon de construire un identifiant est de choisir une suite de caractères tirée au hasard (le plus souvent des chiffres décimaux), dépourvue de toute signification (ce qui exclut de coder en utilisant des nombres successifs car le code prendrait alors une signification ordinale) et en vérifiant que cette suite n'a pas déjà été utilisée. Elle doit contenir assez de caractères pour qu'il soit possible d'identifier tous les individus de la population concernée compte tenu des cohortes qui la renouvelleront pendant le cycle de vie du système d'information c'est-à-dire pendant quelques dizaines d'années. Il est utile enfin de lui associer une clé de contrôle permettant de contrôler l'exactitude de l'identifiant.

* *

La qualité d'une nomenclature se juge, au plan formel, selon l'exactitude du découpage de la population qu'elle fournit : il faut qu'elle constitue une suite de partitions emboîtées, sans omission ni double emploi ; au plan fonctionnel (ou sémantique), selon la pertinence du découpage : il faut que les classes regroupent les individus en fonction de la similitude des actions que l'entreprise entend conduire envers eux ; au plan pratique, selon la clarté de la documentation qui l'accompagne : une nomenclature non commentée, même si elle est pertinente, sera souvent mal interprétée par ceux qui l'utilisent et ils commettront des contresens ; au plan technique, enfin, par la clarté du code utilisé pour identifier les classes de la nomenclature (on utilisera souvent pour désigner un niveau de la nomenclature le nombre de chiffres que contient un code numérique), par les procédures introduites dans les systèmes de saisie ou dans les interfaces pour vérifier la qualité du codage, par la disponibilité des tables de passage (transcodages) assurant la traduction d'une nomenclature dans une autre lorsque cela s'impose (par exemple pour échanger des données avec un partenaire).

Le codage est utilisé dans le système d'information à deux fins distinctes : il peut déterminer le traitement du dossier de l'individu dans la procédure opérationnelle (en qualifiant une demande d'emploi par un métier, en classant un contribuable dans une tranche d'imposition etc.) ; il sert à produire des statistiques sur la population étudiée, chaque individu étant compté dans la classe à laquelle le codage l'affecte.

Les nomenclatures utilisées en pratique peuvent présenter des défauts : si une partie du domaine visé n'est pas couverte par la codification, il y a omission ; si une même partie du domaine peut être classée de deux façons différentes, il y a double emploi (cela peut se produire si les libellés des postes de la nomenclature sont ambigus) ; si le découpage ne correspond pas à l'action que le système d'information est chargé d'outiller, la nomenclature n'est pas pertinente ; si le code est mal défini, il peut provoquer des erreurs etc.

Lorsque deux entreprises entendent faire communiquer leurs systèmes d'information, elles doivent établir des tables de passage entre leurs nomenclatures. Tout est simple si l'on peut établir une bijection entre classes, le passage se ramenant alors à une traduction entre terminologies. Mais le plus souvent il sera impossible d'établir cette bijection car la correspondance se fait entre parties de classes. Dans ce cas la table de passage ne peut être qu'approximative. Il peut en résulter dans le traitement des dossiers individuels une incertitude telle, ou de telles impossibilités, que l'on se sentira obligé de réformer les nomenclatures, tâche lourde et contrariante. Ainsi la vérification de l'adéquation des nomenclatures est le préalable obligé de tout partenariat ou de toute coopération commerciale lorsque le projet implique, comme c'est désormais toujours le cas, de faire coopérer les systèmes d'information.

Évidemment les nomenclatures évoluent : pour rester pertinentes elles doivent pouvoir refléter des priorités changeantes. Cela pose plusieurs problèmes :

- pour assurer la continuité sémantique de la description, le suivi historique des populations devra opérer un transcodage entre versions successives de la nomenclature. Souvent cette tâche est en toute rigueur impossible et l'on ne peut obtenir qu'une approximation ;

- il importe, lorsqu'une nomenclature est changée, que ce changement soit répercuté immédiatement dans les composantes du système d'information qui l'utilisent ; c'est la question de la *gestion des données de référence*. On préfère parfois, pour éviter des à-coups qui rendraient le système d'information instable, prolonger la durée de vie d'une nomenclature un peu au delà de ce qui serait convenable en termes de pure pertinence.

Les nomenclatures sont utilisées par diverses composantes du système d'information (« applications ») et elles évoluent dans le temps même si l'on cherche à les stabiliser (le découpage géographique varie selon l'évolution du marché ; la nomenclature des produits doit être modifiée quand l'entreprise développe un nouveau produit etc.).

Il importe que les tables de codage utilisées par les diverses composantes du système d'information soient mises à jour sans délai : sinon, on risque des erreurs dans l'interprétation des données transmises d'une composante à l'autre, et on risque aussi de produire des statistiques dont l'interprétation sera impossible ou fallacieuse.

La synchronisation des tables de codage ne peut être obtenue que si elles sont asservies à une table mère dite « table de référence ». Il faut que dans la composante qui utilise une table de codage figure le dispositif permettant

d'assurer en continu l'identité entre la table locale et la table de référence : soit cette dernière sera consultée au coup par coup ; soit elle sera répliquée sans délai perceptible dans toutes les composantes qui l'utilisent.

Une erreur fréquente est de « faire comme si » la mise à jour allait de soi et de recopier des composantes contenant une version de la table de référence, mais non les procédures qui permettront de la tenir à jour. Tout se passe bien lors de la recette d'une nouvelle application (puisque la version utilisée est récente) et l'erreur reste inaperçue. C'est dans l'exploitation, de façon sournoise, que l'écart se creusera progressivement entre les tables et que la qualité des données se détériorera (voir page 79).

7.5 Bases de données

La construction des applications informatiques a été grandement facilitée par la mise au point de deux techniques : les systèmes de gestion de bases de données (SGBD, Elmasri et Navathe [144]), qui ont mis de l'ordre dans le stockage et la recherche des données ; les moniteurs transactionnels, qui assurent la communication entre le système et ses utilisateurs.

Les bases de données sont ainsi aujourd'hui l'un des piliers de l'informatique. Les principes selon lesquels elles sont construites devraient pouvoir être aisément compris, quelle que soit par ailleurs la difficulté de leur réalisation technique. Or le vocabulaire qu'utilise la profession gêne leur compréhension. On peut même estimer dangereuse la façon dont les bases de données sont présentées. Certes, quelqu'un de rigoureux et d'intelligent peut toujours s'en sortir sans dommages, de même qu'un funambule bien exercé sait marcher sur un fil. Mais les autres risquent de tomber.

* *

Lorsqu'on considère le monde de la nature et que l'on veut en bâtir une représentation, il faut choisir comme nous l'avons dit les populations que l'on veut voir figurer dans celle-ci (« les clients », « les produits », « les salariés » etc.) et, simultanément, celles dont on fera abstraction. Il faut ensuite définir la façon dont on identifiera les individus qui appartiennent à ces populations, puis les attributs qui seront observés sur chacun d'entre eux.

Mais la théorie des bases de données n'évoque pas ces « populations » ni la façon dont il convient de choisir celles que l'on observera : elle suppose que ce choix a été fait avant que l'on n'entreprenne de construire la base de données. Or nous avons vu qu'il n'allait pas de soi et qu'il pouvait être erroné : nombre d'entreprises ont par exemple longtemps été réticentes à considérer la population de leurs clients, comme le prouve le fait que souvent les clients ne sont pas identifiés.

Il en est de même en ce qui concerne les attributs. Elmasri et Navathe définissent ainsi les données (p. 4) : « known facts that can be recorded and

that have implicit meaning¹² ». Certes, il ne faut pas se battre au sujet de la définition d'un terme aussi général et d'ailleurs, comme l'a dit Aristote lui-même, « il ne faut pas chercher à tout définir¹³ ». Cependant quand une définition est énoncée on peut tenter de voir l'orientation qu'elle indique.

Ici, la définition ignore que (1) la liste des attributs que l'on observera résulte d'un choix qui ne va pas plus de soi que celui des populations, et qui peut lui aussi être erroné (on observe des attributs inutiles, on n'observe pas ceux dont on aurait besoin) ; (2) la façon dont on code les attributs résulte d'un autre choix (pour un attribut qualitatif il faut définir une classification, et les critères formels ne suffisent pas pour définir la « bonne » classification) : un codage mal conçu peut gêner le discernement que le raisonnement, ou l'action, réclament ; (3) la donnée elle-même, telle qu'elle est notée dans la base de données, résulte soit d'une observation dont elle enregistre le résultat, soit d'un traitement appliqué à des observations : or une observation peut être exacte ou erronée, de même qu'un traitement.

* *

Résultant d'une cascade de choix et de conventions, la base de données est une représentation abstraite de la réalité. Dire cela, c'est une banalité mais il est trop tentant, quand on est accaparé par le travail technique, de l'oublier en faisant « comme si » la représentation fournissait de la réalité une image fidèle et sur la pertinence de laquelle il n'y aurait plus à s'interroger. Or même si les choix qui fondent la base de données ont été pertinents lors de sa construction à la date d , rien ne garantit qu'ils puissent l'être encore à une date $d + t$ car la réalité que l'on entend décrire aura changé, ainsi que les relations que l'entreprise souhaite entretenir avec elle.

Boydens [25] a montré, dans son étude de la base de données de la sécurité sociale belge, que les évolutions de la réglementation ou des métiers peuvent altérer la pertinence des codages ou encore obliger à des transcodages rétroactifs qui par nature ne peuvent jamais être exacts - alors que la base de données est utilisée pour déterminer des droits et des taxes, ce qui engage une responsabilité juridique !

L'étude du fonctionnement pratique d'une base de données ne peut donc pas se limiter à la vérification de quelques règles formelles : il faut savoir comment les codes ont été choisis, comment les codages sont réalisés, comment sont prises les décisions qui en résultent, comment enfin les codages sont révisés pour maintenir leur pertinence.

Si l'on a en tête le souci de la pertinence, si l'on sait - non seulement intellectuellement, mais pratiquement - que la définition de la base de données résulte de plusieurs choix, si l'on ne se limite pas à sa définition technique et si l'on considère aussi la façon dont elle est utilisée, alors on ne peut pas se contenter de vérifier sa « normalité ». Sans doute les choses deviennent alors un peu compliquées, alors qu'il est rassurant de se retrancher derrière le rempart du formalisme.

12. « Des faits connus, qui peuvent être enregistrés et ont une signification implicite ».

13. Dans sa *Métaphysique*, Aristote a ainsi renoncé à définir ce qu'est un *acte* ; pour s'en expliquer il a préféré recourir à une liste d'exemples.

C'est sans doute pourquoi la méthode la plus courante de conception d'une base de données omet implicitement les questions sémantiques (Kent [100]) pour ne considérer que les contraintes formelles auxquelles les données doivent obéir. Cette omission n'a rien de critiquable en principe, puisque cette méthode ne prétend pas plus considérer la sémantique qu'un langage de programmation ne considère l'adéquation des applications aux besoins des utilisateurs. Le problème, c'est qu'ici l'omission soit *implicite* : dans la plupart des ouvrages consacrés à la théorie des bases de données (et, je le suppose, dans la plupart des cours), non seulement les questions que pose le choix des populations et la pertinence des codages sont ignorées, mais elles ne sont même pas mentionnées, ce qui incite le lecteur inexpérimenté à les croire négligeables.

On doit enfin distinguer parmi les données : (1) les identifiants, (2) les données observées, puis entrées (« saisies ») dans le système d'information, (3) les données que le système d'information produit en traitant les données observées. Chacune de ces catégories relève d'une gestion spécifique. Certes la théorie des bases de données reconnaît cette distinction¹⁴, mais sans y insister suffisamment puisque souvent les informaticiens ont tendance à manipuler les données comme si elles constituaient une matière pondéreuse indifférenciée analogue à un charbon que l'on traite en masse et en volume, ce qui les empêche de distinguer les catégories de données et de les traiter différemment selon leur rôle.

* *

Le vocabulaire des bases de données comporte des termes pour désigner la façon dont on décrit l'individu type d'une population, mais ces termes varient selon le modèle considéré : on dira « type d'entité » dans le modèle entité-association, « classe » dans le modèle à objets, « relation » dans la base de donnée relationnelle ; dans ces trois modèles, l'individu lui-même sera représenté respectivement par une « entité », un « objet » et un « tuple ».

Observons au passage qu'il faut dire « modèle entité-association » et non « modèle entité-relation ». Lorsque les auteurs de la méthode Merise ont traduit *entity-relationship model* par « modèle entité-relation », ils nous ont fait prendre le risque d'une confusion entre cette relation-là et celle dont il est question dans le SGBD relationnel et qui est tout autre chose.

Dans le modèle entité-association en effet, *association* est comme dans le langage courant ce qui relie deux individus. Par exemple : « Jean-Pierre Martin » (individu) « possède » (association) « le piano Pleyel n° 134878-80 » (individu). Par contre dans la base de données relationnelle *relation* est pris dans un sens qu'il a paraît-il en théorie des ensembles et désigne le *tableau* qui contient, pour une liste d'individus appartenant à une même population et représentés chacun par une ligne appelée « tuple », les valeurs des attributs que l'on a choisi d'observer sur cette population, chaque attribut étant représenté par une colonne¹⁵.

14. Elle parle d'identifiant unique, d'attribut observé, d'attribut dérivé etc.

15. Il peut aussi désigner un tableau contenant la description d'un ensemble d'asso-

Tout cela se range à peu près bien dans la tête une fois qu'on l'a compris, mais le fait que le mot « relation » soit utilisé parfois à la place du mot « association », qu'on lui fasse ainsi jouer deux rôles différents selon le modèle que l'on utilise, ne facilite pas la communication avec les personnes inexpertes.

* *

L'algèbre relationnelle, utilisée pour mettre en forme les calculs qu'on réalise sur une base de données relationnelle, comporte beaucoup plus d'instructions applicables aux attributs que d'instructions applicables aux individus. Il en est de même des « formes normales » qui condensent les contraintes formelles auxquelles une base de données doit obéir. On ne mentionne qu'en passant le fait que les données rassemblées dans un « tuple » (une ligne de la relation) sont relatives à un même individu.

Si l'on avait suivi la démarche qui part de la population pour aller *d'abord* vers l'individu, puis *ensuite* vers les attributs, on aurait abordé la relation non par colonne (par attribut) mais par ligne (par individu). Certes la qualité d'un codage se vérifie attribut par attribut : pour chaque attribut, le codage doit être conforme au type requis. Mais on doit aussi vérifier la cohérence de certaines données individu par individu (exemple : exactitude des additions), et pour calculer les intervalles de vraisemblance nécessaires au repérage des anomalies il faut évaluer des corrélations, ce qui nécessite de considérer la co-occurrence des valeurs prises par les divers attributs dans chaque individu.

* *

Le fait que l'on accorde plus d'importance aux attributs qu'aux individus se manifeste clairement lorsqu'on considère les identifiants, que l'on appelle « clés » dans le modèle relationnel. Selon la théorie il doit exister dans chaque relation un attribut, ou une combinaison d'attributs, qui permette d'identifier chaque tuple de façon univoque.

Cela peut marcher, et cela marche si c'est bien fait, mais cette formulation invite à l'erreur. Certes l'identifiant n'est, au plan formel, qu'un attribut parmi les autres ; mais il joue un rôle tellement important que l'on ferait mieux de l'isoler pour le traiter de façon particulière : un identifiant erroné, c'est un dossier perdu, parfois sans remède ! Par ailleurs il est périlleux de prendre pour identifiant un attribut existant ou une combinaison d'attributs, même s'ils permettent d'identifier chaque tuple de façon univoque : qui sait si, la population évoluant, des homonymes ne risquent pas de survenir ? et alors le caractère univoque de l'identification serait brisé.

On peut tolérer sans doute que des attributs soient pris comme identifiants lors de la discussion entre les informaticiens et les utilisateurs, car c'est plus intuitif : on nommera une entreprise par sa raison sociale, une personne par son état-civil. Mais lorsqu'on passera au modèle logique puis au

ciations : par exemple, un tableau intitulé « possession » dans lequel on trouverait les attributs « possédant », « possédé », « date de début » et éventuellement « date de fin », pour décrire l'association entre des propriétaires et les biens qu'ils possèdent.

modèle physique, il sera impératif de prendre pour identifiant un matricule dépourvu de signification et construit de façon aléatoire.

* *

Dans le modèle relationnel, on note « null » un attribut quand (1) cet attribut n'aurait pas de sens pour l'individu considéré (une personne qui n'a pas d'emploi ne peut pas avoir de téléphone professionnel) ; (2) il aurait un sens mais on ne sait pas s'il existe (cette personne a un emploi, mais on ignore si elle a un téléphone professionnel) ; (3) il aurait un sens mais il n'existe pas (elle a un emploi, mais elle n'a pas de téléphone professionnel) ; (4) il existe, mais on ignore sa valeur (elle a un téléphone professionnel, mais on ne connaît pas le numéro). Je ne suis pas sûr d'avoir fini la liste des significations possibles du code « null ».

Elmasri et Navathe disent (p. 343) que l'on a tenté de construire des SGBD qui distingueraient les diverses acceptions de « null », mais comme c'est compliqué on y a finalement renoncé : rien n'est sorti des nombreuses thèses de doctorat consacrées à ce sujet, modèles probabilistes à l'appui. Il s'agit là d'un point technique, moins crucial que les précédents, mais il reste une faiblesse des SGBD relationnels car à chacune de ces acceptions correspondent des contraintes de gestion différentes. Et certaines bases de données sont aussi désertes qu'une plage anglaise en plein hiver, remplies (si l'on peut dire) de champs vides que l'on ne sait comment interpréter...

7.6 Lisibilité du système d'information

On peut dire qu'un système d'information est « lisible » s'il est *visible* pour ses utilisateurs et, de plus, *compréhensible*. Mais souvent le système d'information est invisible comme l'est l'air que nous respirons (voir page 150).

Dans l'entreprise, chacun utilise le système d'information à sa manière, du président-directeur général à l'agent opérationnel, puisque tout le monde utilise au moins la messagerie (voir page 395). Ici, c'est sur l'agent opérationnel que nous allons concentrer notre attention. C'est lui qui, en pratique, assure la production, les contacts avec les clients, la commercialisation etc. C'est pour qu'il puisse recevoir l'assistance de l'automate que l'entreprise réalise le plus gros de son investissement dans le système d'information : les grandes applications informatiques, qui sont la partie la plus technique du système d'information, sont conçues pour lui.

L'agent opérationnel, c'est le conseiller de l'ANPE, le conseiller clientèle dans une agence bancaire, l'opérateur d'un centre d'appel, le gestionnaire de ressources humaines, le comptable, l'ouvrier en mécanique générale devant sa machine à commande numérique etc. C'est celui qui doit agir sur le terrain, et pour cela réfléchir et décider.

L'entreprise contemporaine délègue de plus en plus la décision opérationnelle aux agents de terrain eux-mêmes car cela lui permet d'accélérer le cycle de production et de gagner en compétitivité. Il en est résulté pour ces agents un accroissement, parfois pesant, de la responsabilité. C'est une des

raisons pour lesquelles le taylorisme n'est plus de mise¹⁶ : on ne peut plus se contenter de pousser à fond la division du travail, puis de spécialiser chaque agent dans une tâche étroite comme dans l'industrie des années 1920. Pour que sa décision puisse être judicieuse, il faut que l'agent opérationnel puisse en anticiper les conséquences. Ceci implique qu'il connaisse la finalité et qu'il comprenne le fonctionnement du processus dans lequel il intervient.

* *

Tout agent opérationnel est impliqué dans divers processus de production. Dans chaque processus, il reçoit une tâche à faire, l'exécute, puis transmet le résultat de son travail à un ou à plusieurs autre(s) agent(s) (figure 7.2).



FIG. 7.2 – *Traiter les dossiers*

Le travail qu'il exécute dans le système d'information se résume à trois actions : lire, écrire, lancer des traitements (figure 7.3).

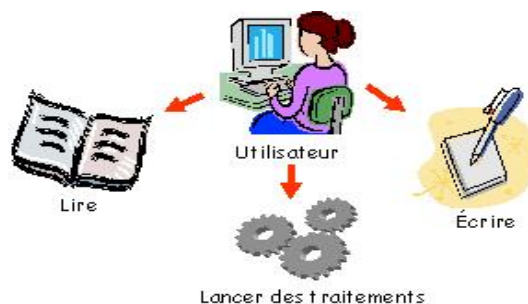


FIG. 7.3 – *Lire, écrire, lancer des traitements*

On pourrait en rester à cette définition pauvre et organiser un taylorisme de l'information : chacun n'aurait à faire, de façon répétitive, qu'une tâche très limitée. C'est là une tentation, car ce taylorisme serait contraire à la décentralisation des responsabilités et des décisions évoquée plus haut et qui est devenue la règle dans l'entreprise d'aujourd'hui.

16. La critique du taylorisme est devenue banale, mais le culte de la division du travail dont il est l'expression est encore vivace alors que les formes contemporaines d'organisation devraient lui imposer de sévères limites.

Pour que système d'information soit lisible - ou, ce qui revient au même, pour que le processus de production soit lisible - il faut que chaque agent opérationnel puisse connaître la réponse aux questions suivantes :

- Quelle est la nature de l'événement qui a déclenché le processus (réception d'une commande, d'une lettre de réclamation, d'une demande de crédit, d'un appel d'offre etc.)?
- Quelle est la nature du client?
- Quel est le produit auquel le processus doit conduire?
- Si ce produit est un livrable intermédiaire, quel est le produit final auquel il contribue?
- Quelle est ma tâche particulière au sein de ce processus?
- Quelles peuvent être les conséquences de mes décisions?
- Comment ma tâche particulière s'articule-t-elle avec celle des autres agents qui interviennent dans le processus, notamment avec ceux qui sont immédiatement en amont et en aval de ma propre activité?
- Quelles sont les exigences de qualité auxquelles le processus doit obéir (délai, satisfaction du client, performance etc.)?
- Quels sont les indicateurs dont on dispose pour vérifier le respect de ces exigences?
- Quelle est ma contribution personnelle à la qualité?

La qualité du processus sera surveillée par un administrateur, ou animateur, qui doit vérifier aussi la consommation des ressources (notamment la charge de travail des agents opérationnels), repérer les incidents, animer leur « debriefing », proposer ou réaliser des modifications du processus. Des statistiques quotidiennes, des alarmes, seront en outre fournies au *manager* opérationnel qui assure l'encadrement immédiat des agents.

Lorsqu'on organise le système d'information autour du processus, on rompt avec l'organisation antérieure en « applications » pour faire en sorte que l'interface fournisse à l'utilisateur, à tout instant, exactement la vue dont il a besoin sur le système d'information (figure 7.4) : cela implique qu'on le soulage de la navigation d'une application à l'autre, des identifications et authentifications répétées, et qu'on supprime les doubles saisies qui en sont le corollaire. Le droit de l'agent à lire, écrire et lancer des traitements est délimité par ses habilitations : la lisibilité du processus, sa « transparence », ne se séparent pas de la gestion de la confidentialité.

Fournir à l'utilisateur, à chaque instant, exactement les plages de consultation et de saisie ainsi que les outils de traitement dont il a besoin, faire en sorte que l'interface évolue pour s'adapter aux diverses étapes de son activité, c'est lui fournir l'outil le plus simple et le plus commode. Pour le système d'information et pour la plate-forme informatique qui le supporte, c'est cependant là une ambition très élevée qu'il est, en pratique, difficile de réaliser entièrement : elle constitue plutôt une orientation qu'un but que l'on puisse atteindre.

Les diverses modes qui se sont succédées en matière d'architecture informatique visaient toutes implicitement cet objectif vers lequel elles ont progressé à tâtons : qu'il s'agisse du client/serveur des années 1990, du couple

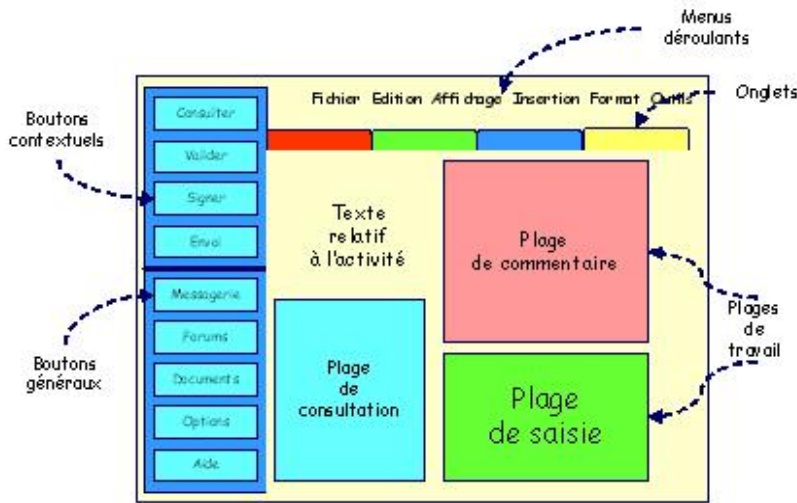


FIG. 7.4 – Vue sur le système d'information

Java-CORBA (langage orienté objets associé à un « middleware ») d'hier, ou aujourd'hui des Web Services et de la SOA (« Services Oriented Architecture »), on ambitionne toujours de fournir à l'utilisateur, à chaque instant et de façon continue, un accès « sans coutures » aux ressources dont il a besoin.

Le système d'information qui y parviendrait, et qui serait donc au sommet de l'état de l'art, paraîtrait cependant à l'utilisateur très naturel et, au fond, tout simple... L'utilisateur, naturellement ingrat, ne percevra pas les difficultés que l'informatique aura dû surmonter pour lui offrir cet outil si « simple », pour en assurer la performance et le fonctionnement sans panne.

Gestion du synchronisme et de la concurrence, physique des bases de données et des moteurs transactionnels, dimensionnement des processeurs, mémoires et réseaux, protection de la sécurité, reprise en cas d'incident : ces questions, qui relèvent de la technique informatique, sont pour l'utilisateur comme les réglages du moteur de son automobile. Il s'attend à ce que ce soit bien fait, il protestera en cas d'incident, mais il ne voit pas *comment* c'est fait parce que cela supposerait des concepts, un vocabulaire différents de ceux qu'implique son action alors que celle-ci est directement concernée et conditionnée par le langage de l'entreprise et par l'organisation du processus.

Il en résulte que la lisibilité du système d'information pour l'utilisateur n'est pas la même que la lisibilité pour la maîtrise d'ouvrage ou pour la maîtrise d'œuvre. La maîtrise d'œuvre doit, bien sûr, avoir une claire visibilité sur les solutions adoptées pour la plate-forme ; et la maîtrise d'ouvrage doit (1) exprimer auprès de la maîtrise d'œuvre ses exigences concernant les performances de la plate-forme ; (2) obtenir la garantie crédible que ces performances seront réalisées, ce qui suppose de comprendre assez bien la technique (voir page 423).

Les clients, fournisseurs et partenaires sont eux aussi des utilisateurs finals du système d'information. Les clients l'alimentent (par exemple en remplissant des formulaires sur la Toile, en utilisant le paiement électronique) et entrent en contact avec l'entreprise *via* divers canaux (téléphone, courrier, Internet, face à face). L'entreprise doit donc concevoir un système d'information qui assure l'unicité de la relation avec le client, quel que soit le canal de communication que celui-ci emprunte : le client s'attend en effet, quand il contacte l'entreprise en face-à-face, *via* la Toile, par téléphone ou courrier, à ce que les indications fournies lors de ses contacts antérieurs soient connues de son interlocuteur.

Aux fournisseurs, partenaires et gros clients réguliers l'entreprise doit fournir sur son système d'information une fenêtre délimitée par des habilitations strictes, mais très nette. Cette exigence s'exprime en termes d'*interopérabilité*. Cela suppose que les systèmes d'information soient de part et d'autre de bonne qualité, que les partenaires aient assuré la cohérence du langage et la transparence du processus au moins pour les tâches partagées.

* *

Lorsque le langage est cohérent et que sa pertinence a été assurée, lorsque les processus ont été organisés et sont familiers aux agents opérationnels, l'entreprise elle-même est lisible à travers son système d'information : elle est *élucidée*, autrement dit elle rayonne sa propre lumière.

Quand on parcourt l'entreprise élucidée, équipée d'un système d'information lisible, on entend des phrases comme celles-ci : « on sait ce qu'on a à faire », « on est bien dirigés », « le boulot est simple », « le travail est intéressant, on sait à quoi on sert » etc. L'entreprise, la direction générale, sont orientées vers la production, l'efficacité, la satisfaction des clients ; les priorités ont été explicitées lors de la définition du langage et des processus ; ce sont les problèmes pratiques du métier, et non plus les conflits entre dirigeants, qui alimentent les conversations à la cantine. Les processus ne sont pas des mécanismes définis voici longtemps, dont on a oublié la finalité et que l'entreprise fait fonctionner par habitude : ils incorporent une démarche collective, à finalité pratique, orientée vers la production et la satisfaction du client.

Lorsque l'agent opérationnel devra s'adapter à une nouvelle situation - nouveau produit, nouvelle technique de production, nouvelle forme de commercialisation, nouveau marché - il saura qu'il faut modifier les processus, puisque leur finalité a changé. Il connaît l'étendue des décisions qui relèvent de lui et sait anticiper leurs conséquences. Il connaît donc les responsabilités qu'il peut prendre, et elles ne l'écrasent pas. Il connaît les indicateurs de qualité : ils lui sont présentés sous une forme graphique évidente qui l'incite à les maintenir à un bon niveau.

Alors l'entreprise ne rencontre plus ces obstacles au changement qui résultent des conflits entre baronnies et entre corporations, ou d'une adhésion rigide à des habitudes. Elle est naturellement évolutive.

Comment obtenir la lisibilité? D'abord en *urbanisant* l'entreprise, en *modélisant* les processus, en instaurant une administration des données, en installant les indicateurs, en nommant des administrateurs de processus. La matière première de la lisibilité consiste tout simplement en un système d'information bien conçu: chacun sait qu'il n'est pas facile d'y parvenir, mais c'est possible si on en a la volonté.

Mais il ne suffit pas de disposer d'un système d'information bien conçu: il faut encore que sa présentation soit éditée sous une forme convenable, lisible. La publication des diagrammes UML ne répond évidemment pas à cette exigence (voir page 323).

Certaines entreprises utilisent des outils qui leur permettent de produire, au dessus de l'urbanisation et des modèles de processus, une couche éditoriale qui confère la visibilité au système d'information. Ces outils illustrent les processus sous la forme de dessins animés (cf. le produit OnMap de Nomia, www.onmap.com): les agents y sont représentés par de petits personnages, placés dans des locaux et devant des matériels semblables à ceux de l'entreprise, qui produisent des documents et échangent des messages. La simulation graphique est accompagnée de commentaires qui s'ouvrent à la demande, ainsi que d'un outil de formation permettant à chacun de contrôler son niveau de connaissance.

La formation ainsi dispensée doit être renouvelée dans la durée: elle accompagne l'activité des agents. Si chacun des segments parmi les utilisateurs finals (voir page 394) dispose d'une « vue » spécifique sur le système d'information, l'outil qui leur fournit la lisibilité doit être décliné selon chacune de ces vues, la cohérence entre elles résultant du fait qu'elles donnent toutes à voir un seul et même système d'information. Ces vues doivent être tenues à jour pour suivre son évolution, et de façon suffisamment synchrone pour écarter le risque d'incohérence.

On peut enfin évaluer la lisibilité, telle qu'elle est perçue par les utilisateurs, en les interrogeant à l'occasion d'enquêtes périodiques de satisfaction relatives au système d'information.

* *

La lisibilité est possible, elle est opportune, elle apporte à l'entreprise efficacité et évolutivité. Le besoin est donc évident. Pourtant, nous le savons, la plupart des systèmes d'information ne sont pas lisibles. Il est impossible de rendre lisible un système d'information mal construit, celui d'une entreprise dont le langage éclate en divers dialectes et qui n'a pas organisé ses processus; mais même l'entreprise qui dispose d'un système d'information convenable ne pensera pas toujours à le rendre lisible:

- on croit souvent que le système d'information est une affaire technique et on néglige la dimension sémantique de l'entreprise;
- dans certaines entreprises on croit encore que l'organisation doit être taylorienne pour être efficace alors même que l'on demande aux agents initiative, décision et prise de responsabilité;
- il arrive aussi que la solidarité que susciterait la lisibilité entre l'entreprise, ses produits, ses clients, ses agents opérationnels contrarie

les baronnies qui militent, au contraire, pour le cloisonnement de l'entreprise et l'éclatement de son langage et préfèrent éviter de laisser apparaître leurs performances dans des indicateurs ;

- l'approche du système d'information par les processus, enfin, se heurte à l'héritage d'une informatique qui, pour des raisons historiques parfaitement fondées, s'est organisée d'abord autour des algorithmes, puis autour des données, alors que l'organisation autour des processus suppose que l'on se concentre sur la construction et le cycle de vie des objets qui représentent les êtres avec lesquels l'entreprise est en relation.

Aucun de ces obstacles n'est insurmontable. Les enjeux de la lisibilité s'expriment en terme d'efficacité, de qualité, de justesse des décisions opérationnelles, d'évolutivité. Certaines entreprises ont compris leur importance.

Il en est résulté un changement de leur attitude envers le système d'information. Alors que celui-ci semblait auparavant technique, abstrait, loin de la pratique et difficilement visible, il est devenu le proche et fidèle compagnon de la pratique professionnelle. Les discussions relatives à sa définition et à son évolution en ont été transformées. L'expression des exigences est devenue raisonnable, sélective et stable. Les compromis nécessaires sont atteints plus aisément, ce qui favorise encore l'évolutivité et la souplesse de l'entreprise.

Chapitre 8

Outils et architecture

8.1 Langages de programmation

Tout programme est écrit dans un langage qui, traduit par un compilateur ou interprété par un interpréteur, pourra ensuite être exécuté automatiquement par l'ordinateur. Dès 1967 on dénombrait 120 langages dont seuls quinze étaient vraiment utilisés (Sammet [178], p. vi). Les programmeurs (que l'on nomme désormais « développeurs », le mot programmeur ayant je ne sais pourquoi été jugé péjoratif¹) utilisent encore aujourd'hui des langages créés dans les années 1950, période d'innovation intense.

Certains langages ont été conçus pour le calcul scientifique, d'autres pour la gestion des entreprises, d'autres pour la formalisation du raisonnement ou le calcul algébrique. Il existe aussi des langages étroitement adaptés à une finalité technique précise.

* *

Au tout début, dans les années 1940, les programmeurs devaient écrire dans le langage machine de l'ordinateur. Son vocabulaire est constitué de nombres binaires représentant les adresses des mémoires et les codes des opérations. Programmer dans ce langage est pénible car il offre peu de repères à la mémoire et à l'intuition de l'être humain.

L'assembleur, conçu en 1950, permet de coder les opérations en utilisant des caractères alphabétiques (ADD pour l'addition, SUB pour la soustraction etc.) et traduit ces codes en langage machine. Néanmoins il était nécessaire de définir des langages encore plus commodes, dits « de haut niveau » (ils sont relativement faciles à apprendre et à utiliser, mais leur mise en œuvre sur la machine suppose une cascade de traductions assurant une optimisation de l'utilisation des couches basses de la machine). Le premier

1. Sans doute a-t-il été victime, avec d'autres, de la « political correctness » qui, remplaçant le nom d'une profession par un autre jugé plus prestigieux, fait comme le maladroit qui élargit une tache au lieu de l'effacer. En 1949 on ne disait pas « programmeur » mais « codeur » : « We were not programmers in those days. The word had not yet come over from England. We were "coders." » (Hopper [85] p. 7).

langage « de haut niveau » fut Fortran (« Formula Translation ») conçu par John Backus à IBM en 1954 (Backus [8]). Ses instructions ressemblent à des formules de mathématiques et il est bien adapté aux besoins des scientifiques, mais incommode pour les travaux peu mathématiques et notamment pour programmer des logiciels de gestion. IBM considérait Fortran comme un langage « propriétaire » qui devait être utilisé uniquement sur ses machines. Algol a été développé en 1958 par un consortium européen pour le concurrencer.

Le Cobol (« Common Business Oriented Language », développé en 1959 par un consortium comprenant le Department of Defense américain) était destiné aux logiciels de gestion. Délibérément « verbeux », le Cobol emploie des mots et une syntaxe proches de l'anglais courant.

D'autres langages encore plus commodes furent introduits ensuite : Basic (« Beginner's All-Purpose Symbolic Instruction Code », 1964) peut être rapidement maîtrisé par le profane ; il est utilisé dans les écoles, entreprises et par les ménages.

C (1972) est un langage de haut niveau mais il peut aussi être utilisé comme un assembleur car il permet de programmer des instructions au plus près de la « physique » de la machine. Beaucoup de logiciels pour les entreprises seront écrits dans ce langage souple dont l'utilisation est dangereuse pour le débutant : comme il permet de tout faire, il comporte peu de « garde-fous ».

Pascal (1970), langage « structuré » conçu de façon à éviter les erreurs de programmation en encourageant la modularité, sera largement utilisé par les pédagogues qui veulent initier les étudiants à la programmation.

Certains langages de haut niveau sont adaptés à des applications précises : APT (« Automatically Programmed Tools ») pour le contrôle des machines outils numériques, GPSS (« General Purpose Simulation System ») pour la construction des modèles de simulation, LISP² pour manipuler des symboles et des listes (suites de symboles) plutôt que des données. LISP sera souvent utilisé en intelligence artificielle ; Scheme est, parmi les dialectes de LISP, celui qui rassemble le plus de partisans (Bloch [13]).

Perl³ est un langage de commande commode dans le monde Unix et pour les serveurs Web. Les langages de quatrième génération (4GL), utilisés surtout pour la gestion et l'interrogation des bases de données, seront encore plus proches du langage humain. On peut citer Focus, SQL (« Structured Query Language ») et dBase.

Les langages à objets (que l'on appelle aussi « langage orientés objets ») comme Simula (1969), Smalltalk⁴ (1980), C++⁵ (1983) ou Java⁶ (1995) permettent d'écrire des logiciels fondés sur des « objets », ce mot désignant de petits programmes qui rassemblent quelques données et traitements et qui communiquent entre eux par des messages (voir page 322). La logique

2. « List Processing », créé par John MacCarthy au MIT en 1959.

3. Créé par Larry Wall en 1987.

4. Créé par Alan Kay (Hiltzik [82]) au PARC de Xerox.

5. Créé par Bjarne Stroustrup (Stroustrup [196]) aux Bell Labs.

6. Créé par Scott McNealy (Stiller et Marlowe [126]) chez Sun.

des langages à objets est proche de celle de la simulation. Leur évolution est allée vers la simplicité et la sécurité : Java contient plus de « garde-fous » que C++, qui comporte les mêmes risques que C.

Les programmes peuvent s'articuler autour d'un « bus » qui assure notamment le routage des messages et soulage d'autant l'écriture de chaque programme : on parle alors d'EAI (« Enterprise Application Integration ») ou encore de « middleware » (voir page 294).

* *

Le mot « programmation » recouvre des activités diverses : l'utilisateur individuel « programme », même s'il ne s'en rend pas compte, lorsqu'il utilise Excel et Word ; il peut aussi, s'il a un tempérament de bricoleur, faire de petits programmes en Pascal ou en Scheme : dans la plupart des cas ce bricolage n'ira pas très loin, même s'il est ingénieux. Les gros programmes sont écrits par des équipes de programmeurs spécialisés qui se partagent les tâches et utilisent souvent des générateurs de code (comme Rational Rose) pour la partie la plus mécanique du travail d'écriture et, pour la documentation et la gestion de configuration, des outils puissants qui facilitent l'évolution du code.

La différence entre un programme individuel et un gros programme est du même ordre que celle qui existe entre le travail (éventuellement réussi) qu'un bricoleur bien équipé peut réaliser à domicile et la construction d'une automobile ou d'un avion, qui suppose une maîtrise de plusieurs techniques dont seule une très grande entreprise peut être capable. Le programmeur individuel ne peut pas rivaliser avec ces usines à programmes, pas plus que l'automobiliste individuel ne peut rivaliser avec un constructeur automobile. Cependant la programmation est pour l'intellect un *hobby* des plus intéressants et des plus formateurs.

L'art de la programmation fait l'objet d'une abondante littérature, qu'il s'agisse d'indiquer des règles de bon sens mais qui ne sont faciles ni à découvrir, ni à mettre en pratique (Arsac [6]), d'introduire à l'informatique à travers la maîtrise d'un langage (Abelson et Sussman [197], Bloch [13], Felleisen et coll. [57]), d'explorer le monde des algorithmes (Knuth [105]) etc.

8.2 Systèmes d'exploitation

Nous avons pris l'habitude barbare de vivre au milieu d'objets techniques dont nous ignorons tout : beaucoup d'entre nous ignorent comment marchent leurs automobile, téléviseur, réfrigérateur, téléphone etc.

Cela ne les empêche pas de s'en servir - ces appareils sont d'ailleurs munis des interfaces qui permettent de les utiliser sans les comprendre - mais cette ignorance a des inconvénients. L'expertise étant le monopole des experts, ils pourraient en abuser : alors un fossé de méfiance se crée et des fantasmes naissent. Savons-nous si le téléphone mobile est mauvais pour la santé ? Si les antennes sont dangereuses pour les habitants des immeubles

où elles sont perchées? Non, sans doute; notre ignorance va si loin que nous sommes incapables de lire les rapports d'expertise, mais cela ne fait qu'amplifier nos émotions.

Parmi ceux qui préfèrent Linux à Windows, ou l'inverse, quelques-uns ont de bonnes raisons mais pour la plupart c'est une affaire de mode ou d'habitude. L'émotion, la mode, l'habitude font de nous des moutons de Panurge. Ne devrions-nous pas étudier les objets familiers qui nous entourent pour pouvoir entretenir avec eux des rapports raisonnables, sinon rationnels?

* *

Pour savoir comment marche notre ordinateur il faut avoir compris ce que fait son système d'exploitation. Les livres sur ce sujet sont arides mais il méritent une lecture attentive: citons par exemple Bloch [14], Raymond [172] et Tanenbaum [199].

Que savons-nous du système d'exploitation? qu'il affiche les fenêtres que nous voyons à l'écran et qu'il gère le répertoire des fichiers (ceux qui ont utilisé MS-DOS dans les années 1980 se rappellent encore les instructions qui permettaient de les créer, déplacer, détruire, concaténer etc.). Nous savons en outre que les systèmes d'exploitation des serveurs centraux sont dotés de fonctions plus riches que celles-là, mais que nous ignorons. Ici s'arrête d'habitude notre science, car la seule partie du système d'exploitation qui soit visible par l'utilisateur est l'interface homme-machine et le répertoire des fichiers. Mais le système d'exploitation exécute aussi pour notre confort, sans que nous n'en sachions rien, nombre de tâches essentielles et discrètes.

Notre ordinateur obéit à l'architecture de von Neumann (1945): un processeur, comportant l'unité de commande, l'unité arithmétique, l'unité d'entrée sortie et des registres, puise dans la mémoire les instructions et les données; il consulte ces dernières, les traite, puis charge dans la mémoire le résultat des traitements. Lors de l'exécution d'un programme, le système d'exploitation reçoit les instructions que fournit le compilateur, écrites sous la forme d'une suite de 0 et de 1 qui représentent des commandes et des adresses. Il déclenche l'exécution d'autres programmes qui gèrent les processus en cours, la mémoire et les équipements périphériques (disque dur, écran, clavier, imprimante etc.). Son « noyau » (« kernel ») regroupe tous les programmes qui interagissent avec les éléments matériels de l'ordinateur. Ne retenons ici, à titre d'exemple, que deux des fonctions qu'a décrites Laurent Bloch: l'exécution pseudo-simultanée des programmes et la gestion de la mémoire.

Exécution pseudo-simultanée

Le processeur exécute les instructions de façon séquentielle, l'une après l'autre; pourtant nous utilisons plusieurs applications simultanément (sont ouverts en ce moment sur mon PC FrontPage, Word, Outlook Express, Internet Explorer et Group Mail Pro: oui, je suis un des captifs de Microsoft...) Faire exécuter les programmes de façon pseudo-simultanée, cela revient à conférer au processeur la capacité d'arrêter le programme en cours, de gérer les priorités dans la file d'attente des programmes candidats, de relancer l'exécution d'un programme antérieurement arrêté.

Considérons une opération des plus simples comme déplacer le curseur en fin de ligne lors de la saisie d'un texte. Si nous utilisons un raccourci clavier, cette action prendra 0,25 secondes et se traduira par environ 20 000 instructions élémentaires. Si notre processeur a une vitesse de 500 MIPS, il peut en 0,25 secondes traiter 124,97 millions d'instructions supplémentaires. 20 000 instructions élémentaires, c'est donc pour lui très peu de chose...

La mesure des performances

On a d'abord mesuré la performance par l'inverse du temps moyen nécessaire pour exécuter une instruction, ou MIPS (million d'instructions par seconde). Mais lorsque les architectures se sont diversifiées en jeux d'instructions différents les MIPS n'ont plus permis de comparer les performances.

Des *benchmarks* permirent de définir la notion de « MIPS relatif » : si par convention le VAX-11/780 était un ordinateur à 1 MIPS, on pouvait dire qu'une machine cinq fois plus rapide avait un débit de cinq MIPS relatifs.

En 1977, une mesure de performance fit cependant apparaître que le débit du VAX-11/780 était seulement de 0,5 MIPS originels. Cela permit de dire que les VAX sur lesquels on avait constaté 3 MIPS originels étaient des ordinateurs à 6 MIPS relatifs... Au début des années 1980, le terme MIPS désignait presque toujours des MIPS relatifs.

Les supercalculateurs des années 1970 et 80 avaient de hautes performances sur des programmes utilisant de façon intensive le calcul en virgule flottante. On définit alors pour évaluer les *benchmarks* une nouvelle unité, le MFLOPS (million d'opération flottantes par seconde). Mais le discours commercial ne retint pour comparer les ordinateurs que les MFLOPS crête, et non ceux qu'indiquaient les *benchmarks*.

La mesure des performances était ainsi entourée d'une confusion propice aux manipulations. Pour permettre des comparaisons plus fiables et des benchmarks plus réalistes, le SPEC (*System Performance and Evaluation Cooperative*) fut créé à la fin des années 1980 (Hennessy et Patterson [153] p. 73).

Certains systèmes d'exploitation « non préemptifs » comme MS-DOS ou les premières versions de Windows ne géraient pas la file d'attente des programmes : ce soin était laissé aux programmes eux-mêmes. Certains programmeurs trouvaient cela « cool » : comme c'était décentralisé ils croyaient que c'était « démocratique » : la démagogie existe aussi en informatique. Mais cela provoquait des pannes fréquentes, notamment si le PC travaille en réseau ou s'il se produit des interruptions asynchrones (provoquées par un élément matériel extérieur au processeur, elles peuvent se produire alors qu'une instruction est en cours de traitement).

Gestion de la mémoire

Comme le coût de la mémoire est fonction croissante de la vitesse d'accès, le PC utilise plusieurs mémoires dont la taille est d'autant plus petite que la vitesse d'accès est plus élevée (le disque dur, mémoire la plus volumineuse, est aussi la plus lente). Outre la RAM, où résident les programmes en cours d'exécution, le processeur comporte des mémoires caches très petites (quelques Mo) et très rapides, dont le système d'exploitation assure la gestion (fort délicate).

La « mémoire virtuelle » est une technique d'écriture des adresses : les adresses virtuelles que comporte le programme ne sont traduites en adresses réelles que quand c'est nécessaire. Cela permet d'utiliser un espace d'adressage aussi grand qu'on le souhaite et de se libérer des contraintes liées à la taille de la mémoire physique. La traduction des adresses virtuelles en adresses réelles est effectuée à la volée par la MMU (« Memory Management Unit »). Les programmes, eux, n'ont à connaître que les adresses virtuelles. L'appel d'une donnée par un programme doit pouvoir être indifférent à la nature de la mémoire sur laquelle elle réside : c'est au système d'exploitation qu'il revient d'automatiser les entrées et sorties.

8.3 Découpage en « applications »

Tout système informatique est un dispositif dans lequel des données (résultats de mesures effectuées sur des faits du monde réel, ou événements) sont :

- *entrées* (soit elles sont saisies, soit elles proviennent d'autres programmes),
- *traitées* (addition, calcul de ratios etc.) afin de procurer de nouvelles données, les résultats.

Les résultats sont édités (mis en mémoire ou présentés sur une interface) soit pour commander un automate, soit pour être utilisés par un être humain qui décidera et agira en conséquence (figure 8.1).

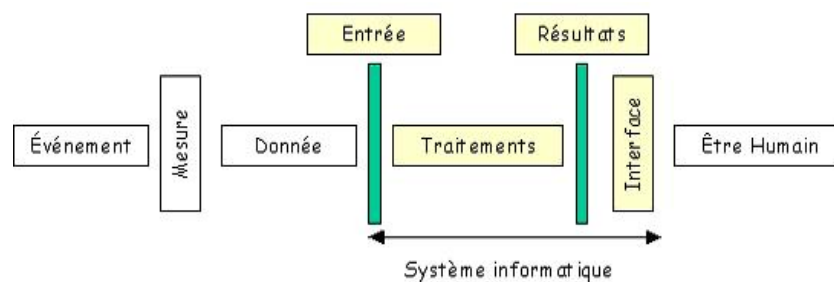


FIG. 8.1 – Place du système informatique dans le système d'information

La complexité des traitements interdit en général de maîtriser un système global. On le découpe donc en applications, exécutées chacune par un

programme particulier. On peut modéliser ce découpage en disant qu'il faut construire un tableau croisant données d'entrée et résultats, puis le classer de façon à lui donner une structure bloc-diagonale (beaucoup d'échanges à l'intérieur d'un même bloc, peu d'échanges entre blocs différents) : à chacun de ces blocs correspond une application (figure 8.2).

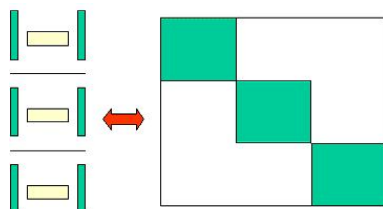


FIG. 8.2 – *Délimitation des applications*

Logiciel, application, logiciel système, progiciel

Parmi les logiciels, on a été amené à distinguer les « logiciels systèmes » qui fournissent des ressources utilisées par les applications (système d'exploitation de l'ordinateur, pilotes, BIOS ^a, moniteur transactionnel, système d'exploitation du réseau, protocoles de communication etc.).

À l'origine, le mot « application » a été retenu pour désigner une utilisation spécifique de l'ordinateur en vue d'automatiser l'une des fonctions de l'entreprise comme la gestion des stocks, la paie ou la facturation : l'ordinateur est en effet alors « appliqué » à cette fonction.

Parmi les applications, on distingue celles qui nécessitent un programme spécifique et celles qu'il est plus efficace de faire réaliser par un logiciel du commerce (figure 8.3), appelé « produit logiciel » ou encore « progiciel » (SGBD, traitement de texte, tableur, logiciel graphique, messagerie, agenda et annuaire téléphonique, gestion de projet, CAD, production d'images, progiciels mathématiques et scientifiques, production multimedia, navigateurs Web etc.).

^a « Basic Input/Output System ».

Le système informatique est alors défini par un ensemble d'applications à l'intérieur desquelles les échanges de données sont intenses mais dont les échanges mutuels sont relativement faibles. Ce découpage, associé à une organisation des échanges « transverses » et des référentiels, c'est la tâche de l'urbanisme. Elle se fait en pratique de façon empirique et approximative car elle doit tenir compte de contraintes qui résultent de l'histoire de l'entreprise (de même, quand on organise l'entreprise, il convient en principe de regrouper dans les mêmes unités les personnes qui ont beaucoup d'informations à partager ou à échanger ; cette répartition se fait de façon empirique, la théorie servant seulement d'orientation).

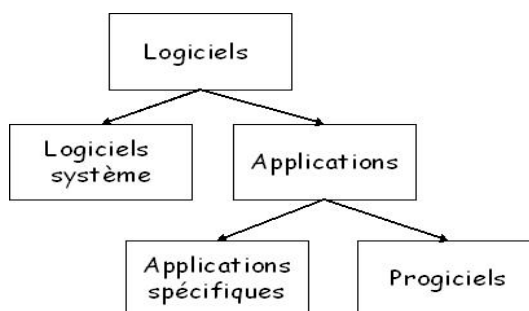


FIG. 8.3 – Classification des logiciels

Les applications reçoivent chacune un nom propre (« Chéops », « Clémentine », « Facturation » etc.). Elles sont représentées par une icône sur le poste de travail des utilisateurs, ce qui leur confère une sorte de personnalité. Elles sont définies selon les besoins des utilisateurs (de même, on construit une maison selon les besoins de ses habitants futurs), mais une fois construites elles ont une existence propre et tout se passe comme si l'utilisateur devait s'adapter aux applications : il en est de même pour quelqu'un qui emménage dans une maison existante.

Pour réaliser l'application une fois qu'elle a été définie, l'attention se concentre sur les données et les traitements, sur la partie informatique du système d'information et sur les contraintes techniques auxquelles celle-ci doit obéir : cela ne doit pas faire oublier que le système d'information est fait pour être utilisé, tout comme une maison est faite pour être habitée.

* *

Pour construire un programme, il suffit de définir les données et de programmer les traitements, ou « algorithmes » (d'où le titre du livre de Niklaus Wirth, créateur du langage Pascal : *Algorithms + Data Structures = Programs* [219]). Mais sauf dans les cas les plus simples on ne peut pas plus se lancer directement dans la programmation que l'on ne pourrait construire une maison sans avoir dessiné son plan.

Le plan d'un programme, c'est un *modèle* qui doit correspondre aux besoins des utilisateurs tout en tenant compte de l'état de l'art informatique et des contraintes techniques et économiques de la réalisation. Les principales qualités sémantiques que doit posséder un modèle sont les mêmes que celles que l'on exige du système d'information (voir page 255) :

- la *pertinence* : il faut que le modèle réponde correctement aux besoins des utilisateurs ;
- puis la *sobriété* : comme toute représentation du monde réel, celle que donne le modèle est schématique ; ce schématisme doit non seulement être accepté, mais assumé de façon positive : il faut que le système d'information soit le plus simple possible ;
- enfin la *cohérence* : si l'application comportait une incohérence elle serait fautive et il faudrait corriger cette « bogue » ; elle doit en outre être

cohérente avec le système d'information de l'entreprise : ses identifiants et ses nomenclatures doivent suivre les évolutions du référentiel.

Historiquement, l'attention des informaticiens s'est d'abord concentrée sur les traitements ; ils ont mis au point des algorithmes (Knuth [105]) qui permettent de calculer (l'approximation des nombres réels par des nombres rationnels pose de redoutables problèmes mathématiques, voir la note page 83), produire des nombres au hasard (nécessaires pour les calculs qui simulent un comportement aléatoire), classer et trier les données etc.

Les informaticiens ont aussi défini des structures de données, du plus simple (chaîne de caractère, booléen, entier relatif, rationnel, réel) au plus compliqué (vecteurs, tenseurs, structures composites). La connaissance des structures et algorithmes classiques, ainsi que de leurs conditions d'utilisation, fait partie du bagage professionnel de l'informaticien qui doit maîtriser aussi les procédés de vérification des programmes (« débogage »).

Outre les qualités sémantiques citées ci-dessus, l'application doit avoir des qualités techniques : elle doit être construite et documentée de telle sorte qu'une modification introduite en un endroit du programme ne provoque pas d'erreurs dans d'autres parties du programme, que la correction des « bogues » soit aisée, enfin qu'il soit facile d'y introduire des changements pour tenir compte de l'évolution des besoins des utilisateurs (« évolutivité »).

L'évolutivité implique et résume toutes les autres qualités techniques : pour que l'application soit évolutive, il faut que les modifications ne provoquent pas d'erreur et qu'il soit aisé de corriger les erreurs. La sobriété favorise l'évolutivité : toutes choses égales d'ailleurs, plus une application est simple, plus il sera facile de la faire évoluer.

Enfin, le système informatique doit assurer les échanges de données entre l'application et les autres applications en assurant le synchronisme et les transcodages nécessaires (voir page 294).

Une application peut fonctionner parfaitement et satisfaire les utilisateurs lors de sa recette sans être pour autant ni sobre, ni cohérente, ni évolutive ; ces défauts n'apparaîtront qu'à la longue et ils entraîneront un coût de maintenance élevé et une obsolescence rapide. Après quelques années il faudra la refaire et, dans l'attente de la réfection, les utilisateurs auront dû se servir d'un outil inadapté. Le manque d'évolutivité accroît donc le coût annuel moyen de l'application tout en altérant sa pertinence.

Lorsqu'on corrige un logiciel, sa taille s'accroît et sa qualité se dégrade ; il en résulte une obsolescence technique qui se combine à l'obsolescence fonctionnelle pour réduire la durée de vie des applications (voir page 526).

8.3.1 Les ERP

Martin Campbell-Kelly [29] propose une classification du logiciel en trois catégories :

Logiciel sur mesure : cette catégorie recouvre les développements réalisés pour les entreprises par des SSII, sur cahier des charges, en réponse à des spécifications précises et propres à l'entreprise considérée.

Produits logiciels (assemblages ou en anglais *packages*) : ces logiciels sont analogues à des biens d'équipement ; ils sont produits par des entreprises spécialisées et vendus en petit nombre (et donc cher) aux entreprises qu'ils équipent. Parmi les produits logiciels, on distingue ceux qui sont destinés à une secteur d'activité (assemblages pour l'architecture, la finance, la pharmacie etc.) et ceux qui concernent une fonction de l'entreprise (comptabilité, gestion des ressources humaines, gestion des approvisionnements etc.).

Progiciels grand public : ce sont des biens de consommation destinés au grand public, aux utilisateurs de micro-ordinateurs (systèmes d'exploitation, suites bureautiques, jeux). Ils sont vendus en grand nombre à un prix très inférieur à celui des produits logiciels.

Historiquement, les premiers logiciels ont été produits sur mesure. Puis les fournisseurs ont cherché, en réutilisant pour d'autres clients les solutions conçues pour un premier client, à rentabiliser l'expérience acquise. Ils ont ainsi mis au point des assemblages (*packages*) qu'ils ont d'abord utilisés en interne mais qu'il a fallu documenter de façon de plus en plus précise, notamment pour les mettre entre les mains des personnes nouvellement recrutées.

Ces produits ont ainsi mûri et ils sont enfin arrivés au degré de qualité, d'« industrialisation » (documentation, maintenance), qui permettait de les vendre aux clients eux-mêmes moyennant une formation des utilisateurs.

L'ERP se trouve tout à l'extrémité de cette évolution : c'est, par contraste avec les assemblages sectoriels ou fonctionnels, un assemblage destiné *a priori* à tous les secteurs, à toutes les fonctions, les adaptations nécessaires se faisant par paramétrage.

* *

Le premier ERP a été mis au point par SAP, entreprise allemande dont l'acronyme se développe en allemand comme en anglais : « Systeme, Anwendungen, Produkte in der Datenverarbeitung » ou « Systems, Applications & Products in Data Processing ».

SAP a été créée en 1972 à Walldorf, en Bavière, par cinq anciens programmeurs d'IBM Allemagne. Comme toutes les petites entreprises de logiciel celle-ci a commencé par produire du sur mesure. Son premier contrat portait sur un système comptable et financier en temps réel pour l'usine d'Imperial Chemical Industries (entreprise britannique) à Ostragen.

À l'époque peu de programmeurs étaient capable de réaliser un système en temps réel et la petite équipe de SAP s'est ainsi placée à un haut niveau d'expertise. Conformément à la stratégie qui vise à réutiliser et rentabiliser l'expertise accumulée sur un premier contrat, SAP a construit à partir de cette solution un produit interne, le « System R », progressivement amélioré et enrichi lors des contrats suivants de façon à devenir de plus en plus universel tout en restant intégré et cohérent.

L'idée de base de SAP a été d'organiser le système d'information autour d'une base de données unique, alimentée et utilisée par les diverses applications. La cohérence du système est garantie par l'unicité de la base

de données : il est exclu qu'une même information soit représentée par plusieurs données différentes.

En 1980 SAP est la dix-septième SSII allemande et elle a pour clients la moitié des cent plus grandes entreprises allemandes. Elle est protégée de la concurrence américaine d'une part parce que les entreprises de logiciel américaines ne s'intéressent pas encore au marché européen (elles ont à cette époque trop à faire sur le marché américain), d'autre part parce que le produit de SAP est d'une qualité très supérieure à celle des logiciels américains : la mise au point est soigneuse, réalisée par une équipe de haut niveau rigoureuse et disciplinée : la programmation à l'allemande se distingue du « good enough » empirique à l'américaine.

En 1981, SAP propose R/2, assemblage pour *mainframe*. En 1982, SAP a 250 clients allemands, dont plusieurs filiales allemandes de groupes internationaux. Ces filiales seront pour SAP des chevaux de Troie vers l'internationalisation : lorsque les grands groupes découvriront la qualité du logiciel dont s'est équipée leur filiale allemande, ils voudront le généraliser au monde entier.

SAP crée un bureau international à Genève en 1984 et internationalise R/2, dont le paramétrage devient capable de prendre en compte la diversité des monnaies, législations fiscales etc. Le produit devient ainsi adaptable à tous les pays au prix de l'ajustement de milliers de paramètres.

L'adaptation du progiciel de SAP à une entreprise devient une spécialité parmi les consultants ; la qualification en SAP sera bientôt très recherchée par les SSII et SAP crée en 1987 un Centre international de formation pour les consultants.

Un bureau américain est ouvert à Philadelphie près des premiers clients américains de SAP en 1988. SAP définit pour les États-Unis une politique commerciale spécifique, les vendeurs percevant un intéressement qui rendra jaloux les commerciaux allemands. Des groupes d'utilisateurs sont créés, des partenariats avec les SSII sont montés. En 1990, SAP est au quatrième rang mondial des producteurs de logiciels.

Le système R/3, commercialisé en 1992, est adapté au client/serveur. En 1993, le livre de Hammer et Champy [36] lance la mode du re-engineering qui répond d'ailleurs à une nécessité : les entreprises s'étant équipées d'applications non cohérentes, le besoin d'une réorganisation du système d'information se faisait sentir. L'ERP se présentait comme un ensemble de composants logiciels avec lequel on pourrait construire un système d'information : beaucoup d'entreprises estimeront que la meilleure solution consiste à mettre au rebut leurs anciennes applications et à tout reconstruire autour de R/3.

SAP avait mis au point une nouvelle formule tarifaire : le prix de la licence R/3 dépendait du nombre de postes de travail équipés (2 700 à 4 000 \$ par station). Cela lui permit d'atteindre des niveaux de prix jusqu'alors jamais vus dans le marché du logiciel.

* *

Cependant le coût de la licence ne représente qu'une faible partie de la dépense que l'entreprise doit supporter pour implanter un ERP. Le travail

de paramétrage, réalisé par des consultants, ainsi que la « conduite du changement » - il est souvent préférable de redéfinir les procédures de l'entreprise plutôt que d'adapter l'ERP - font que l'installation coûte de 5 à 20 fois le prix de la licence.

En 1998, on dénombre 20 000 installations de R/3 et 1,5 millions de salariés l'utilisent quotidiennement. Campbell-Kelly estime que SAP joue un rôle crucial dans l'économie mondiale, bien plus important à ses yeux que celui de Microsoft. SAP, qui employait 9 personnes en 1972 et faisait 300 000 € de chiffre d'affaires, emploie 24 178 personnes en 2000 et fait 6,3 milliards d'euros de chiffre d'affaires. La maturation de SAP a duré près de vingt ans : c'est seulement à partir de son installation aux États-Unis en 1988 que la croissance est devenue rapide.

SAP n'a pas pu conserver le monopole des ERP : d'autres entreprises se sont lancées dans les années 1990 sur ce marché prometteur. Baan, entreprise néerlandaise dont les origines et l'histoire ressemblent un peu à celles de SAP, se lance aux Pays-Bas en 1978 et aux États-Unis en 1993. Oracle, fournisseur du SGBD utilisé par la plupart des clients de SAP, a mis au point son propre ERP « Oracle Applications » autour de sa propre base de données et s'est lancé sur le marché en 1995. JD Edwards, Peoplesoft, System Software Associates, entreprises déjà présentes sur le marché du produit logiciel, systématisent leur offre pour offrir des ERP. Par la suite Oracle achètera Peoplesoft.

D'après *JDN Solutions* du 16 janvier 2006⁷, SAP détenait en 2005 45 % du marché en France, contre 22 % pour Oracle et 33 % pour l'ensemble des autres fournisseurs d'ERP (Sage/Adonix, Intentia Consulting etc.) ; en 2004, les parts de marché dans le monde étaient de 40 % pour SAP, 12 % pour Peoplesoft et 10 % pour Oracle.

* *

Les commerciaux des ERP disposent de puissants arguments face aux dirigeants⁸ : ils leur proposent de mettre un terme au désordre du système d'information et leur donnent l'espoir de régler des problèmes d'organisation sans grand effort politique. Cette offre séduisante par sa qualité et sa cohérence est cependant à l'usage plus risquée que l'on aurait pu l'imaginer : elle ne peut être efficace que si l'on accepte les contraintes qu'elle impose, et sa mise en œuvre comporte des difficultés et des pièges.

Si l'ERP incorpore une expertise professionnelle qui fournit à l'entreprise des modèles de processus conformes à l'état de l'art, ses fournisseurs ne peuvent pas incorporer dans leur produit une compétence universelle car cela leur demanderait trop de travail : dans les domaines où l'entreprise est particulièrement « pointue » elle dispose d'une expertise meilleure que celle que l'ERP peut apporter.

Ainsi il sera raisonnable de faire appel à l'ERP pour les fonctions qui ne relèvent pas de son cœur de métier ; par contre sur le cœur de métier où il lui

7. solutions.journaldunet.com/dossiers/chiffres/erp.shtml

8. Guillaume Benci, « Ingénierie du SI à base de progiciel », conférence au Séminaire du club des maîtres d'ouvrage des systèmes d'information, 25 mai 2000.

importe d'être plus efficace que ses concurrents elle ne pourra généralement pas se contenter de l'ERP et devra utiliser un logiciel « sur mesure » réalisé par une SSII sur cahier des charges, puis traiter les problèmes que pose l'intégration de ce logiciel avec l'ERP.

Par ailleurs adopter un ERP implique plus qu'un contrat avec le fournisseur : c'est un *mariage* qui comporte des obligations, et il sera plus difficile d'en sortir que d'y entrer. Si l'éditeur a créé un club d'utilisateurs, l'entreprise devra y faire participer ses propres experts mais cela leur prendra du temps.

Tout ERP a des limites et il est inévitable qu'elles ne coïncident pas avec ce que l'entreprise aurait souhaité. Elle aurait voulu un carré, c'est un losange qui lui est livré (figure 8.4) : il manque des choses à l'ERP par rapport à ses besoins, et il fait par ailleurs des choses dont elle n'a pas besoin.

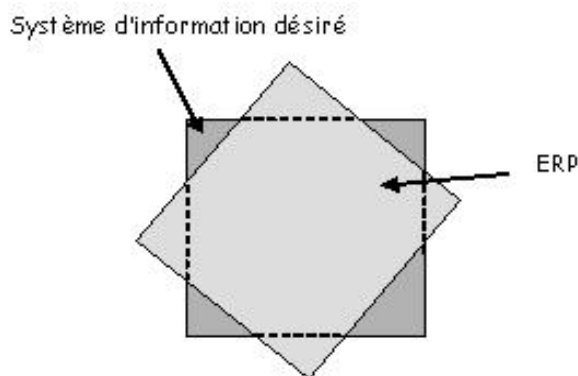


FIG. 8.4 – Écart entre l'ERP et le SI désiré

Les choses que l'ERP fait en trop sont éventuellement gérées dans l'entreprise par d'autres espaces fonctionnels : il faudra traiter les redondances et chevauchements qui en résultent. Ce qui manque, mais qui est nécessaire, devra être fait ailleurs tout en étant cohérent avec l'ERP : cela occasionnera un travail supplémentaire d'architecture.

L'ERP est fourni avec ses propres solutions en ce qui concerne le référentiel (catalogue des produits, référentiel des clients et fournisseurs, inventaire des stocks etc.) : si l'entreprise a mis en place des solutions différentes, elle devra y renoncer car il est moins coûteux de s'adapter à l'ERP que d'adapter celui-ci à l'entreprise.

Il se peut enfin que l'éditeur de l'ERP n'ait pas choisi les mêmes logiciels système que l'entreprise (système d'exploitation, SGBD etc.). Adopter l'ERP peut la contraindre soit à gérer en parallèle plusieurs versions des logiciels système, ce qui implique une duplication des compétences et des équipes au sein de la DSI, soit à se plier aux choix faits pour l'ERP - ce sera le plus souvent préférable, mais au prix d'une réfection de la plate-forme informatique.

L'adoption d'un ERP ne représente pas un seul projet. Le fournisseur publiera des versions successives, et elles seront si différentes les unes des autres que le passage d'une version à la suivante sera un véritable projet. Lors de la sortie d'une nouvelle version, il faut en effet :

- faire l'inventaire de ce qui est proposé, évaluer ce qui est intéressant, choisir ;
- évaluer le coût des travaux de reconception : la « compatibilité ascendante » relève plus du discours commercial que de la réalité et il faudra refaire beaucoup de paramétrages ;
- évaluer l'effet du changement de version sur tout ce qui se trouve à la périphérie du progiciel, et qu'il impacte.

Avant de conclure le « mariage », il faudra prendre des précautions : la négociation du contrat est délicate. Il convient de réaliser d'abord une étude de faisabilité approfondie, et il faudra lutter pour obtenir des informations de l'éditeur avant la signature du contrat. Il faut ensuite vérifier la capacité de l'éditeur à accompagner l'entreprise dans la durée et à partager avec elle son expertise sur le métier.

Certains des enjeux de l'entreprise ne pourront être atteints, si elle adopte l'ERP, qu'à condition de changer la façon dont elle fait son métier. Il faut donc une maîtrise d'ouvrage encore plus forte que lorsqu'on conçoit un logiciel spécifique : de nombreuses demandes d'adaptation de l'ERP vont s'exprimer et il va falloir pouvoir leur résister. Le dialogue avec les responsables des métiers devra être approfondi : si l'ERP ne permet pas de mettre en œuvre les règles souhaitées par le marketing en matière de facturation, il faudra pouvoir s'assurer de l'accord de la direction marketing.

Enfin, comme le mariage est de longue durée, il faut que l'entreprise acquière une compétence sur l'ERP. Lorsqu'une entreprise achète un ERP, elle n'a pas en effet à payer seulement la licence du produit : elle doit aussi s'associer un cabinet de consultants et c'est de loin la dépense la plus importante (dans un cas que j'ai connu, la licence avait coûté un million d'euros mais le coût total du projet a été de vingt millions d'euros). L'entreprise doit se former pour pouvoir être autonome lors des projets qu'elle devra conduire à l'occasion des changements de version.

L'utilisation de l'ERP réussit mieux dans les PME que dans les grandes entreprises, car les PME n'ont pas beaucoup d'argent à dépenser et savent aller droit à l'essentiel. Dans les grandes entreprises, la première cause d'échec est le caractère versatile de la maîtrise d'ouvrage, qui modifie trop souvent son expression de besoins et ses priorités ; la deuxième cause d'échec est le conflit de pouvoir entre maîtrise d'ouvrage et maîtrise d'œuvre informatique.

8.3.2 Les EAI

J'ai vu quelquefois le commercial d'un fournisseur ou d'une SSII présenter une offre d'EAI (« Enterprise Application Integration ») à des dirigeants ou à des DSI. Il appuyait son argumentation par la projection d'un transparent qui ressemblait à la figure 8.5 :

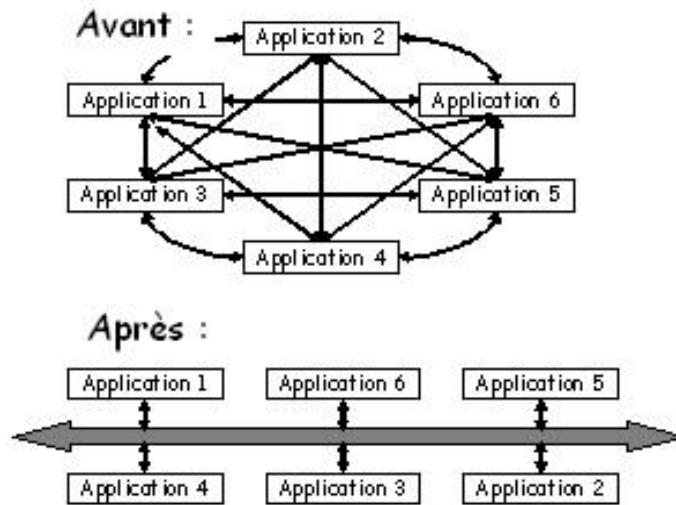


FIG. 8.5 – Les EAI d'après le discours commercial

« Avant », c'est le « plat de spaghetti » : chaque application doit gérer sa connexion avec chacune des autres. « Après » (avec l'EAI), chaque application est connectée à un « bus » et à lui seul. Comme c'est simple ! Le commercial plaide la cause de l'EAI avec la conviction d'un missionnaire lancé en pleine évangélisation. « Qui, s'exclame-t-il, préférera le plat de spaghetti ? » « Pas moi en tout cas », répondent en chœur le dirigeant et le DSI.

Chacun bien sûr préfère que le système d'information soit en ordre. Mais par quel miracle ce « bus » si simple arrive-t-il à remplacer le « plat de spaghetti » d'autrefois ? S'il y parvient, c'est parce qu'il n'est pas simple du tout. En fait il ne s'agit pas vraiment d'un « bus », support passif de transmission, mais d'un *routeur* équipé de logiciels qui assurent activement des fonctions de commutation et quelques autres encore. Les soi-disant « bus » EAI assurent, entre autres fonctions, celles par exemple que définit la norme CORBA (« Common Object Request Broker Architecture ») :

- routage : le bus interprète l'étiquette du message pour identifier l'adresse vers laquelle il doit l'orienter ;

- transcodage : si deux applications recourent à des codages différents, le bus assure la traduction entre les codes ;

- gestion des flux : en puisant dans une gamme de synchronismes qui s'étend du temps réel transactionnel au traitement *batch* qui est réalisé en fin de journée, le bus gère les délais de mise à jour selon les besoins des applications. Il comporte des files d'attente (*buffers*). Il traite la concurrence (lorsque deux utilisateurs veulent modifier en même temps une même donnée) et la persistance (lorsqu'il est nécessaire de garder en mémoire la valeur d'une donnée qui vient d'être modifiée).

Les fonctions offertes par le bus soulagent d'autant les applications : pour ne prendre qu'un exemple, il est plus simple pour une application d'envoyer les messages au bus qui en assurera le routage que d'inclure elle-même un sous-programme de communication.

D'ailleurs on peut connecter au bus des « services » (applications spécialisées dans la fourniture d'un service aux autres applications : sécurité, supervision, métrologie etc.). Il en résulte une économie d'échelle par concentration du code naguère dispersé dans plusieurs applications.

On demande aussi parfois au bus de présenter le système d'information sur l'interface homme-machine de telle sorte que l'utilisateur dispose exactement, et « sans couture », des « vues » sur le système d'information qui lui sont nécessaires (données, espaces de saisie, traitements).

Le bus EAI est en définitive un « middleware » très riche en fonctionnalités techniques. Sa mise en œuvre suppose que les fonctions ci-dessus soient définies et paramétrées. L'utilité de l'EAI est réelle - mon propos n'est pas de le dénigrer - mais sa présentation par les commerciaux est d'une simplicité trompeuse. Demandez d'ailleurs au commercial qui cherche à vous vendre un EAI comment son produit gère les routages, transcodages, synchronismes, concurrence, persistance etc. Le plus souvent il n'en saura rien, le pauvre : ses seules armes sont le transparent ci-dessus et son savoir-faire de vendeur professionnel. Une fois le contrat signé arriveront les spécialistes. Pour pouvoir construire le « bus » ils devront poser une liste de questions qui dissipera bientôt la première image, fallacieuse, de simplicité.

8.4 Intranet et Datamining

Selon une étude réalisée en 1996 sur les 500 plus grandes entreprises de Hong Kong (Lai [107]), 41 de ces entreprises avaient mis en place des Intranets. L'étude montre qu'elles ont avancé à l'aveuglette : « Aucune étude quantitative ou qualitative n'a été entreprise pour comprendre comment on doit faire du datamining », « la plupart des études sur la performance des Intranet ont été conceptuelles ou anecdotiques ; aucune n'a tenté de collecter des données empiriques auprès des entreprises qui en ont mis en place, ni utilisé leur expérience en matière de planning, maîtrise, installation et évaluation ». Ceux qui recommandent depuis des années d'utiliser ces techniques ont donc la foi du charbonnier, puisqu'il n'existe pas d'étude des conditions pratiques du succès ; les travaux théoriques sont certes nombreux, mais quelle garantie apportent-ils ?

8.4.1 Datamining

Karim K. Hirji [83] a étudié un projet de datamining réalisé en 1998 pour une entreprise canadienne de restauration rapide. On peut tirer de cette étude quelques enseignements de portée générale.

Le datamining (en français « fouille dans les données ») est une méthode statistique qui vise, à partir de l'examen des données, à induire des modèles descriptifs ou même explicatifs. Il fournit, en réponse à des ques-

tions économiques ou scientifiques, des classifications (« clustering »), des études de corrélation (analyse en composantes principales, analyse des correspondances), des outils de « scoring » (analyse discriminante), des modèles prévisionnels (régression multiple).

Des « arbres de décision » utilisant des « réseaux de neurones » aident à s'orienter dans la forêt des méthodes et dans leur paramétrage. Sur le plan théorique, cela ressemble à de l'économétrie (le modèle est choisi *a priori* par tâtonnement), mais plus souvent encore à l'analyse des données (lorsqu'il n'existe pas de modèle *a priori*): le datamining, c'est en fait l'analyse des données (Volle [212]) appliquée à de grands volumes de données, et portée ainsi au niveau d'une industrie. Dans le cas le plus simple, le datamining aide à trouver l'explication d'une anomalie constatée dans les données statistiques en facilitant la « descente » vers des données détaillées, voire vers un dossier individuel.

Sa mise en œuvre rencontre trois difficultés :

1) qualité des données : 60 à 70 % du travail est consacré à la préparation des données (sélection, correction, transcodage, chargement). Si l'entreprise dispose d'un datawarehouse (voir page 299), les questions de qualité auront en principe été déjà traitées : alors la préparation des données peut ne représenter que 30 % du travail (c'est ce qui s'est passé dans l'exemple considéré) ;

2) choix de l'itinéraire des travaux : les « réseaux de neurones » ne règlent pas tout. Pour pouvoir répondre aux questions que se pose le client, les méthodes doivent être choisies en dialoguant avec lui. Il faut donc que l'expert en datamining soit aussi un animateur et possède des qualités que l'on rencontre rarement ensemble dans une même personne : rigueur dans la méthode, ouverture et chaleur humaine dans la communication ;

3) évaluation des résultats : le datamining confirme des choses que connaît déjà celui qui consacre plusieurs heures par jour à l'examen manuel des données, d'où la déception du client expert (sauf bien sûr si l'on enrichit le datamining par recoupement avec d'autres données qu'il n'aura pas encore utilisées). On peut lui fournir plusieurs réponses :

a) il est heureux que les enseignements fournis par l'examen assidu des données soient confirmés lorsqu'on utilise des outils plus puissants : que dirait le client si ce n'était pas le cas !

b) les « évidences » que le datamining retrouve sont, certes, déjà connues de l'expert attentif, mais il a mis beaucoup de temps à les découvrir et il lui est difficile de les communiquer à ceux qui n'ont pas fait le même travail. Plus rapide, le datamining facilite la communication des « évidences » et permet aussi de les classer par ordre d'importance en leur associant une mesure de leur signification statistique ;

c) si le travail manuel sur les données a déjà fourni l'essentiel des conclusions utiles, c'est qu'il constitue déjà une forme artisanale mais efficace de datamining. Cela conduit à s'interroger sur l'utilité du datamining « industriel », fortement outillé et donc coûteux (l'exemple étudié est de taille modeste - 20 jours de travail - mais s'équiper pour une utilisation systématique du datamining serait une lourde affaire).

Le praticien qui examine les données jour après jour règle en effet chemin faisant les problèmes de sélection, d'estimation des données manquantes, de redressement, de transcodage etc. qui sont le premier obstacle au datamining. Pour interpréter les données, il formule des hypothèses et les teste. Ce travail se fait à petite échelle, sur un tableur, sans prétention théorique, mais il donne des résultats de valeur (la preuve : on les retrouve après une approche « industrielle lourde »).

Il ne faut pas sous-estimer l'efficacité d'un travail quotidien, assidu, qui ronge les obstacles comme les vagues rongent une falaise. Seulement les résultats obtenus par le praticien sont difficiles à communiquer, à partager ; les autres personnes ne peuvent pas savoir s'il s'agit de vrais résultats, sérieux et solides, ou de lubies individuelles.

Le datamining, avec sa lourdeur et son caractère systématique, apporte aux résultats la caution de son poids. Si l'entreprise dispose d'un statisticien crédible, pondéré, qui examine assidûment les données, il apportera armé de son habileté et de son tableur - ou si l'on veut d'un logiciel statistique comme SAS - autant et plus que tous les outils de datamining, à condition qu'il sache communiquer et qu'elle sache l'écouter.

Mais il faut parfois, pour faire passer une « évidence » dans l'entreprise, jouer la comédie de la rigueur ostentatoire en recourant à des outils sophistiqués. Il est vrai aussi que certaines « évidences » ne se révéleront pas si l'on a pour seuls outils un bon cerveau et un tableur : alors les lourdes méthodes du datamining peuvent être utiles. Mais il est prudent, avant de les utiliser, de s'assurer que l'on a déjà tiré des données ce qu'un bon statisticien peut en extraire avec les méthodes « manuelles ».

8.4.2 Intranet

Utiliser un Intranet, c'est utiliser dans l'entreprise des outils analogues à ceux que l'on trouve sur l'Internet : messagerie, Web, forums, et les diverses extensions rendues possibles par Java, Perl et autres XML : formulaires, moteurs de recherche, statistiques d'utilisation etc.

L'Intranet est utilisé d'abord pour la communication interne, notamment la rédaction, la diffusion et la tenue à jour des documentations professionnelles. Lai [107] attribue cette mission à la direction des ressources humaines, mais elle peut être remplie par d'autres directions (communication, production, commerciale).

L'Intranet est un moyen peu coûteux et efficace pour mettre la documentation de l'entreprise à la disposition de l'ensemble son personnel, quelle que soit la localisation de celui-ci dans le monde, car il utilise les standards de l'Internet et de la Toile (TCP/IP, HTTP, HTML) et sa mise en place ne nécessite donc pas de chambouler les réseaux ni les ordinateurs. Pour une petite entreprise, c'est une façon économique de s'informatiser, la petite taille ne constituant pas ici un désavantage. Les apports essentiels sont l'accès des salariés à une information à jour et l'amélioration de la communication sur l'organisation. Les difficultés résident dans la gestion du contenu, l'animation des groupes de travail et la maîtrise des réseaux.

Ce dernier point est important : s'il est possible de déployer un Intranet sur les réseaux existants, le débit qu'il suscitera lorsque son utilisation se développe peut contraindre ensuite à des redimensionnements et à l'acquisition de compétences nouvelles en télécommunications. Les informaticiens doivent eux aussi se qualifier dans les techniques de l'Intranet notamment pour des raisons de sécurité.

Si les entreprises utilisatrices ont constaté un gain d'efficacité dans la communication, le gain serait par contre peu sensible dans le travail coopératif : l'Intranet présente encore des lacunes par rapport à un outil de groupware comme Lotus Notes en ce qui concerne l'intégration avec les bases de données et les applications, ainsi que la sécurité. Des efforts sont faits pour combler cet écart notamment autour de XML. C'est une évolution importante pour l'articulation entre les documents que véhicule l'Intranet et les données qui constituent la matière première du système d'information.

8.5 Le système informatique d'aide à la décision

Le SIAD (Volle [215]; on dit aussi *datawarehouse*) est un outil d'observation et de description qui vise, à partir de données de gestion ou de statistiques, à donner aux dirigeants et aux *managers* d'une entreprise les moyens d'identifier des alertes de gestion, de suivre l'évolution de l'activité, d'examiner des sujets ou phénomènes particuliers. Il ne fournit pas cependant les explications ni les commentaires, qui relèvent d'un travail postérieur à l'observation et à la description.

Il occupe dans l'architecture du système d'information une place très particulière. Alors que tous les autres éléments de cette architecture sont soutenus par certaines parties de l'édifice et en soutiennent d'autres, le SIAD est soutenu par tout le reste mais ne soutient rien lui-même. Il ne contribue pas à la solidité du système d'information mais il est, comme le coq sur le clocher d'un village, posé sur une pointe qu'il rend visible de loin. On peut dire aussi qu'il fait rayonner le système d'information comme s'il était l'antenne d'un émetteur.

Il tire parti de l'ensemble des données produites ou acquises par l'entreprise, ensemble dont il fournit une présentation synthétique. Cela suppose (1) qu'il soit alimenté potentiellement par toutes les applications de l'entreprise, (2) qu'il résolve les problèmes de comparabilité et de redressement des données qui sont inévitables lorsqu'on utilise des sources diverses.

Le SIAD vise à présenter des informations *utiles*. Ceci implique qu'il soit construit de façon sélective en choisissant, parmi les statistiques qu'il serait possible de produire, celles qui peuvent servir à telle ou telle catégorie d'utilisateurs. Sa construction suppose donc une analyse des besoins, elle même fondée sur une segmentation des utilisateurs en sous-populations homogènes en ce qui concerne les missions à remplir et les besoins qui leur correspondent.

Le SIAD vise à fournir aux utilisateurs un outil de consultation comode, d'une ergonomie aisée, de façon à minimiser les tâches de recherche de l'information et de présentation des résultats.

Produire des statistiques en adressant au coup par coup des requêtes à une application opérationnelle serait coûteux en traitement. Le SIAD protège donc les bases de données opérationnelles en s'intercalant comme un tampon entre elles et les utilisateurs et en préparant la plupart des statistiques dont ces derniers ont besoin. Les outils fournis par le SIAD pour remplir ces objectifs sont :

- des tableaux de bord comportant des alertes ;
- des tableaux préformatés contenant l'essentiel de la statistique d'activité et d'environnement ;
- des tableaux mis à disposition selon la technique des « hypercubes » ;
- la publication d'analyses (analyses de corrélation, simulations etc.) utilisant les outils du « datamining ».

Les hypercubes sont des tableaux multidimensionnels intermédiaires qui permettent de construire à la demande, par sélection, des séries chronologiques ou des tableaux croisés. Le contenu des hypercubes doit être défini *a priori*, à partir de l'analyse des besoins, de sorte qu'ils satisfassent au mieux leurs utilisateurs.

Étant sélectif, le SIAD ne peut pas répondre à toutes les questions possibles mais seulement à la plupart des questions. Il peut donc arriver qu'un utilisateur recherche une information que le SIAD ne fournit pas, et il faut pourtant que l'on puisse lui répondre : ce sera la tâche d'une équipe d'analystes habilités à utiliser des requêtes pour interroger la base de données. Toutefois le délai de réponse sera plus long (quelques heures ou quelques jours) que celui de la consultation des hypercubes, qui n'est que de quelques secondes.

L'évolution des hypercubes, comme du dispositif dans son ensemble, sera réalisée par une cellule d'administration centrale en relation avec les analystes.

* *

Un SIAD peut être présenté selon trois couches (figure 8.6) :

- alimentation par les applications opérationnelles ;
- stockage historisé, agrégation et alimentation des hypercubes ;
- publication sous forme d'alerteurs, de tableaux préformatés, de tableaux croisés et de graphiques.

La base de données du SIAD est exploitée par une équipe d'analystes pour produire les hypercubes, et aussi pour répondre à des requêtes complexes envoyées par les utilisateurs. Seuls les hypercubes sont directement consultables par les utilisateurs qui peuvent ainsi construire une grande diversité de tableaux croisés. La constitution de la base de données du SIAD à partir des applications qui l'alimentent nécessite une procédure permettant d'extraire périodiquement de chaque application les données nécessaires. Ces données sont rangées par l'application dans une base de données in-

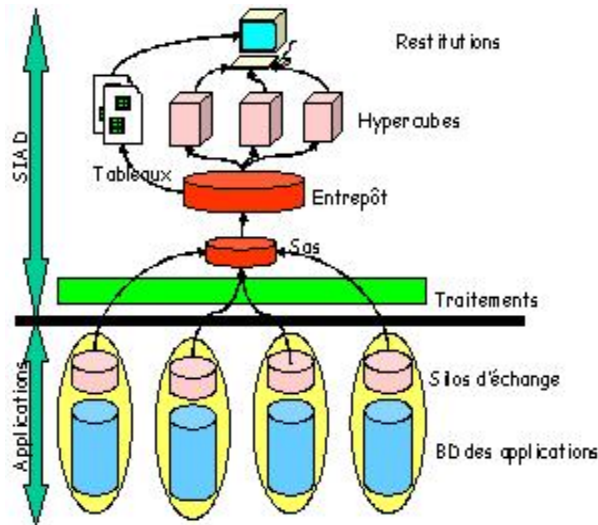


FIG. 8.6 – Les couches du SIAD

termédiaire appelée « silo d'échange » dont la responsabilité appartient à l'application source. Le SIAD doit réaliser ensuite un traitement pour vérifier et redresser les données avant de les intégrer à sa propre base de données, l'« entrepôt de données ».

La structure de l'entrepôt fait l'objet d'une modélisation formelle qui précise les « axes », variables qualitatives dont le croisement définit les tableaux que le SIAD pourra produire (exemple : mois, région, secteur d'établissement etc.), et les « attributs », variables qui seront ventilées dans les cases des tableaux (exemple : nombre de personnes, montants en euros, nombre d'entretiens etc.).

La base de données du SIAD est nécessairement détaillée : lorsque l'application source contient des enregistrements individuels, la base de données est elle-même individuelle. Cependant alors que la base de données opérationnelle est « vivante », c'est-à-dire subit des modifications par mises à jour continues, le SIAD est une base « morte » qui enregistre des situations passées dont elle doit permettre de reconstituer l'enchaînement chronologique. À chaque individu correspondent un identifiant fixe et des données qui évoluent dans le temps. Seule une sélection de ces variables intéresse le SIAD ; celui-ci sera donc à la fois exhaustif (en ce qui concerne les individus composant la population étudiée) et sélectif (en ce qui concerne les attributs observés sur chaque individu).

Les hypercubes sont produits à partir de la base de données du SIAD par agrégation de données individuelles : si le SIAD contient les données détaillées sur les clients, les hypercubes permettront de construire des tableaux décrivant la population des clients.

Il est *a priori* possible d'utiliser deux méthodes différentes pour alimenter l'entrepôt de données : prendre une suite périodique de « photographies »

instantanées de la base vivante, ou considérer les événements qui modifient cette base.

Option 1 : Suite de photographies

Supposons que le SIAD soit alimenté par une copie périodique de la base vivante (par exemple on copie dans le silo d'échange chaque vendredi à 20h00 des enregistrements fournissant, pour chaque client, les valeurs des variables sélectionnées). La dimension historique du SIAD est alors obtenue en considérant la succession de ces enregistrements hebdomadaires. Le volume de l'entrepôt de données croît progressivement par empilement de fichiers hebdomadaires, ce qui peut poser à terme un problème de volumétrie.

Option 2 : Événements

On appelle ici « événement » toute modification d'un enregistrement individuel affectant l'un des attributs sélectionnés pour le SIAD. Entre deux événements, les valeurs de ces attributs ne sont pas modifiées.

Utiliser les événements pour nourrir l'entrepôt, et non des photographies périodiques, apporte deux améliorations : d'une part le volume de l'entrepôt sera plus réduit puisque les enregistrements qui n'ont pas été modifiés ne seront pas recopiés ; d'autre part le fait que les événements soient datés permet de construire lors des exploitations un découpage chronologique quelconque (par semaine, par mois etc.), alors par exemple que le calcul exact de données mensuelles à partir des données hebdomadaires que fournissent les photographies n'est pas possible.

Soulignons que l'on entend par « événement » la modification d'une variable au moins de l'enregistrement individuel considéré (donc d'une variable interne au dossier considéré, qu'elle soit calculée ou obtenue par observation) : ainsi le SIAD n'a pas à reproduire les traitements réalisés au sein de l'application source, dont il recueille les résultats.

Si l'on choisit la deuxième option, l'entrepôt comporte : une photo initiale de la base source, constituée par la liste exhaustive des enregistrements individuels identifiés et, pour chaque enregistrement, la liste sélective des variables observées ; puis des enregistrements datés, correspondant à chaque événement de telle sorte que le traitement de la base permette de reconstruire l'évolution historique de chaque individu.

Nota Bene : pour limiter le volume des données on distingue, parmi les variables observées, celles dont on souhaite suivre l'historique et celles dont on souhaite seulement connaître l'état actuel. Les événements concernant les variables dont on souhaite suivre l'historique sont conservés en mémoire ; pour les autres seule la valeur la plus récente est conservée, et elle « écrase » les valeurs antérieures. Cette distinction se fait à l'intérieur de l'entrepôt de données : elle ne concerne pas le silo d'échange qui doit donc contenir temporairement tous les événements.

* *

À partir de ce qui précède il est aisé de concevoir la nature du silo d'échange que chaque application doit constituer. Au démarrage du SIAD, l'application construit une base de données initiale indiquant pour chaque enregistrement individuel la valeur des variables observées par le SIAD.

Cette base de données sera ensuite traitée pour amorcer l'entrepôt de données.

Puis le silo sera alimenté, selon la technique choisie, soit par les événements, soit par les photographies au fur et à mesure de leur occurrence. Il faut donc prévoir, à l'intérieur de l'application, un mécanisme qui détectera les événements, ou qui prendra les photographies, et enverra les enregistrements correspondants vers le silo où ils seront stockés. Le moteur d'alimentation du SIAD consulte périodiquement le silo, recopie ses éléments vers une base temporaire, le « Sas » (ils seront ensuite traités pour alimenter l'entrepôt), puis le purge.

Le silo est une base de données de taille modeste, son volume se limitant à celui des événements survenus entre deux consultations par le SIAD ou à celui de la dernière photographie. Il peut contenir aussi des indications techniques visant à garantir la qualité de l'alimentation du SIAD ; il faut en effet s'assurer (et ce n'est pas facile) :

- que l'image de l'application figurant dans le SIAD ne diverge pas de la réalité par suite d'un cumul d'erreurs dans la collecte des événements ;
- que des opérations visant à « nettoyer » l'application (purge d'enregistrements désuets, corrections des codages et identifiants) ne suscitent pas des erreurs en provoquant des événements factices ;
- que les modifications des classifications et nomenclatures utilisées dans l'application sont correctement répercutées dans le SIAD.

Commande adressée par le SIAD à une application

On voit maintenant ce que doit contenir la commande adressée par le SIAD à une application :

- définition des « individus » qui seront observés ; *a priori*, tous les individus gérés par l'application intéressent le SIAD (clients et commandes, fournisseurs et offres) ;
- liste des attributs qui seront observés par le SIAD sur chacun de ces individus ;
- indications techniques visant à garantir la qualité de l'alimentation du SIAD.

Les responsables de l'application devront, à partir de cette commande, faire réaliser le logiciel qui permettra d'alimenter le silo conformément à la méthode décrite ci-dessus. Il n'est pas indispensable que le SIAD indique dès la passation de sa commande la liste définitive des variables qu'il voudra observer : une fois le mécanisme d'alimentation du silo d'échange en place, cette liste peut être modifiée car elle est un paramètre de ce mécanisme.

Traitements réalisés par le SIAD

Les données issues d'une application opérationnelle ne se prêtent jamais telles quelles à une exploitation statistique : il faut corriger les erreurs, estimer les données manquantes etc. Entre le silo d'échange et l'entrepôt de données s'intercale donc un traitement des données.

L'existence de ce traitement peut poser problème lors de l'utilisation des données : lorsqu'on remplace une donnée manquante par une estimation, cela peut donner une information utilisable au niveau France entière mais

fausser les proportions au niveau d'une commune ou d'une région. Il faudra donc disposer de contrôles ou d'alarmes garantissant la représentativité des données.

Le traitement comporte deux étapes : la première apporte des corrections purement techniques visant à garantir la qualité statistique des données. L'autre apporte des transcodages visant à assurer la compatibilité des données avec les définitions réglementaires et comptables.

* *

Apport du SIAD à la gestion

Le SIAD a pour but de faciliter l'orientation de l'action en fournissant des données qui alimentent la compréhension du marché et de l'activité de l'entreprise, après recoupement éventuel avec d'autres sources (économiques, démographiques, marketing etc.).

Il fournira donc les indicateurs de pilotage permettant à un responsable opérationnel d'évaluer la qualité et la productivité du travail fourni par des établissements ou des équipes, indicateurs qui impliquent un recoupement avec des données que le SIAD ne comporte pas toujours (volume et qualité des ressources employées, délais de traitement des affaires etc.).

Il n'a pas vocation à fournir des indicateurs pour un pilotage opérationnel au jour le jour ou pour le suivi de dossiers individuels. Il faut donc que chaque processus soit muni des outils permettant aux responsables opérationnels de piloter les travaux au plus près de leur réalisation. Cependant le SIAD peut contribuer à l'alimentation de ces outils : un responsable peut trouver, dans les hypercubes que produit le SIAD, telle série chronologique qu'il recoupera avec des données de gestion pour évaluer l'efficacité du travail de son unité.

Apport du SIAD à l'analyse

Certains représentent l'architecture du SIAD par une pyramide. Sa large base est constituée des applications qui l'alimentent, le sommet par les hypercubes et autres outils d'observations synthétiques (figure 8.7).

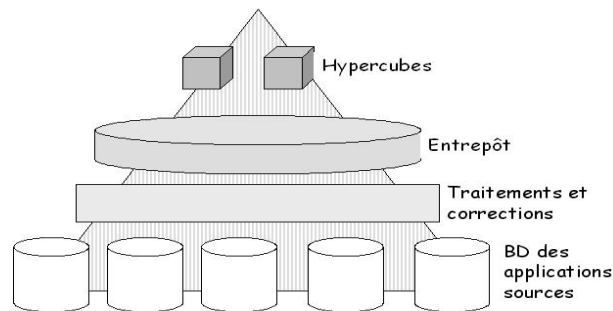


FIG. 8.7 – La « pyramide », représentation répandue mais partielle

Un SIAD est bâti à partir des données d'observation, et il faut distinguer l'observation de l'explication : un microscope permet de voir des bactéries

mais n'explique pas comment elles se reproduisent. Le SIAD appelle donc un dépassement vers l'explication et le commentaire. Ce dépassement implique, pour pouvoir servir les diverses populations d'utilisateurs concernées, une diversification éditoriale.

C'est en complétant le SIAD par des outils d'analyse des données et d'économétrie, en le confrontant aux modèles explicatifs, que l'on pourra l'utiliser pour comprendre ce qui se passe sur un marché.

Il est utile de représenter les opérations éditoriales qui s'appuient sur le SIAD ; utilisant les données d'observation synthétiques, elles permettent de produire des résultats interprétés et commentés destinés à diverses populations d'utilisateurs (responsables régionaux, responsables de ligne de produit etc.). La représentation du SIAD n'a plus alors la forme d'une pyramide mais celle d'un diabolos (figure 8.8).

S'il est souvent nécessaire pour l'interprétation d'utiliser les méthodes de l'analyse des données ou de l'économétrie, il est recommandé de rien en laisser paraître dans la publication qui ne doit recueillir que leurs résultats : leur technicité repousserait le lecteur, seuls les résultats l'intéressent.

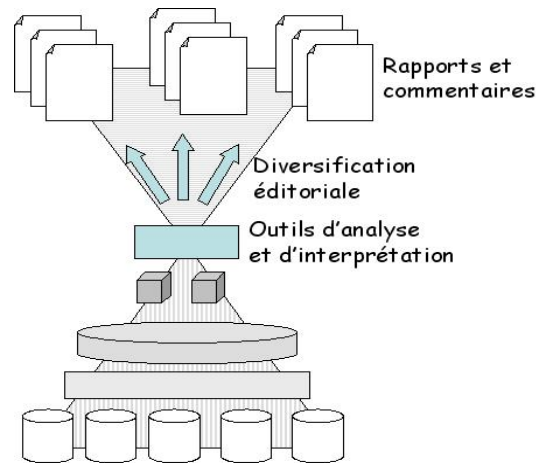


FIG. 8.8 – Le « diabolos », représentation complète du SIAD

8.5.1 Étude de cas : histoire d'un Datawarehouse

L'entreprise Artichico a envisagé, en 1996, de se doter d'un datawarehouse à finalité commerciale. Les sources disponibles étaient des données détaillées fournies par les distributeurs et les commandes des clients. L'enjeu était d'alimenter des analyses : positionnement concurrentiel de l'entreprise ; comportements d'achat ; effets de la fidélisation ; qualité du service ; rentabilité des campagnes de publicité ; segmentation des clients ; ventes en partenariat ; canaux de distribution etc.

Les utilisateurs potentiels du datawarehouse étaient les personnels des directions régionales, ceux qui mettent au point les contrats avec les clients,

ceux qui définissent la politique en matière de prix, les experts du marketing, les économistes etc.

Il existait à Artichico un ou plusieurs infocentres⁹ pour chaque application opérationnelle, mais ils étaient cloisonnés, peu ergonomiques, peu évolutifs, et ne répondaient pas à tous les besoins.

L'exploitation des commandes des clients individuels supposait que l'on sût les identifier. C'est facile si le client est abonné ou inscrit dans le programme de fidélisation ; sinon, il faut estimer son identifiant à partir des *éléments d'identification* le plus souvent incomplets fournis lors de la commande (nom, prénom, nom de l'entreprise, numéro de téléphone, numéro de la carte bleue etc.). La construction du référentiel des clients était donc une affaire délicate.

* *

Un premier projet de datawarehouse, plus verbal que pratique, a échoué : ses spécifications manquaient de précision et l'entreprise n'avait pas accordé les moyens nécessaires pour lancer sa réalisation. Il fut relancé par une autre équipe en 1997. Il a démarré effectivement en 1998, la mise en service débutant en 2002.

Le datawarehouse commercial d'Artichico aura finalement coûté de l'ordre de 25 millions d'euros. Il est destiné à plus de mille utilisateurs internes à l'entreprise. On prévoit de l'ordre de 10 000 connexions par semaine à terme. Il est aujourd'hui considéré comme une des réussites du système d'information de l'entreprise.

La solution technique articule plusieurs progiciels : un pour les extractions et les traitements, un autre pour la production des tableaux, enfin un SGBD. La « base détail » du datawarehouse est alimentée par les applications opérationnelles ; des agrégats sont produits selon les nomenclatures que fournissent les référentiels.

L'exploitation du datawarehouse est une usine informatique qui consomme chaque mois 80 heures de CPU pour le traitement mensuel, 120 heures pour le traitement hebdomadaire, 200 heures pour le traitement quotidien, soit 400 heures en tout. La base de données occupe 2 To et utilise 1 000 tables de référence. Le traitement comporte chaque mois 200 passages de chaînes différentes : il faut les orchestrer, traiter les alertes, réaliser des reprises etc. Pour maîtriser cette usine, il a fallu concevoir pour les opérateurs une « chorégraphie » précisément minutée ; les programmeurs ont dû s'astreindre à produire un code exploitable et bien documenté.

La réalisation

La direction du projet à la maîtrise d'œuvre a d'abord piétiné pendant un an et demi en produisant des développements jetables. Le chef de projet ambitionnait d'organiser la production à la place des opérateurs mais

9. La différence entre datawarehouse et infocentre ne réside pas dans la dimension temporelle (certains infocentres contiennent des séries chronologiques) mais dans le fait que l'infocentre est dédié à une seule source alors que le datawarehouse est alimenté par des sources diverses.

les scripts ne comportaient pas de points de reprise, ce qui impliquait que l'opérateur refit tout son travail depuis le début chaque fois qu'il rencontrait une difficulté.

Il a fallu tout reconstruire sur des bases plus solides après un changement de responsable. Cependant l'informatique s'est encore enlisée en s'efforçant d'assurer une large couverture fonctionnelle tout en stabilisant la technique.

La maîtrise d'ouvrage a exigé en 2001 un moratoire du développement : ne programmer aucun nouveau module tant que les modules existants ne seraient pas en état de marche. Ce moratoire a contraint l'informatique à se concentrer sur la stabilisation de l'existant et l'a mise au défi de réussir.

Les opérateurs ont eu un réflexe de recul devant l'utilisation de progiciels nouveaux. Il a fallu, avec l'accord du DSI, instaurer une relation directe entre eux et la maîtrise d'ouvrage. Elle a été finalement appréciée par les opérateurs et a permis de débloquent la situation. Aujourd'hui la production est en place, le taux d'erreur est nul, les opérateurs sont fiers du succès de l'opération.

Le volume de l'application, conjugué au manque de maîtrise des technologies, a retardé sa mise en œuvre. Ce retard a suscité une perte de confiance des utilisateurs auxquels l'entreprise avait annoncé une mise en place en deux mois. Comme toujours pendant qu'un projet est bloqué, les maîtrises d'ouvrage se sont activées pour peaufiner les spécifications, d'où une sur-enchère des demandes fonctionnelles. En outre la lenteur de la réalisation a incité des analystes à fournir aux utilisateurs des CD-Rom contenant les données retraitées sur les parts de marché : ce n'était qu'une solution partielle, mais elle a occupé le terrain pendant un temps et risqué de démobiliser les énergies consacrées au datawarehouse.

Définition fonctionnelle

Le datawarehouse est alimenté par plusieurs sources¹⁰. L'un de ses apports a été de rendre visibles les défauts de ces applications et d'inciter à les corriger. La correction des données incombe aux applications ; comme la maîtrise d'ouvrage du datawarehouse était aussi celle des applications commerciales, elle a pu assumer elle-même la responsabilité de la qualité de ces sources.

Un datawarehouse qui traite selon diverses périodicités (mensuelle, hebdomadaire, quotidienne) des données concernant 5 millions d'individus est de type industriel. Les données qu'il fournit sont brutes, sans redressement ni correction des erreurs et données manquantes : si une anomalie est détectée, il est demandé à l'application source de la corriger. Artichico tolère dans ses applications opérationnelles jusqu'à 2 % de doublons, cette tolérance s'applique donc aussi au datawarehouse : ce datawarehouse non corrigé vise à assister les décisions opérationnelles. Les économistes qui voudront l'utiliser pour leurs études devront, si nécessaire, corriger eux-mêmes les données.

10. Données sur les parts de marché, offres des entreprises concurrentes, offre de l'entreprise, commande des clients, enquêtes de satisfaction auprès des clients, relations après vente, programme de fidélisation, indicateurs de qualité, contrats avec les distributeurs etc.

Lors de l'élaboration des spécifications, les utilisateurs ont eu tendance à tout demander parce qu'il leur était difficile d'explicitier ce qu'ils attendaient d'un datawarehouse ; le risque était alors qu'une forte proportion des tableaux produits ne fût jamais utilisée.

Pour la version 1, on a donc d'abord défini pour chaque module au plus dix tableaux, puis on a défini l'univers de données qui permettrait de les produire (une fois cet univers défini, il permet de produire aussi d'autres tableaux). On a procédé ainsi à l'inverse de la démarche qui construit d'abord un hypercube puis le met à disposition. Les utilisateurs peuvent, s'ils ont besoin de tableaux non prévus, demander que l'on exécute des requêtes : l'expérience montre que cette possibilité est peu utilisée. Une version 2 sera si nécessaire réalisée après un an pour corriger les lacunes de la version 1.

Le datawarehouse d'Artichico ne comporte pas d'hypercube. Artichico a d'abord travaillé avec un progiciel qui construit des hypercubes mais ce produit n'est plus maintenu et les hypercubes posaient des problèmes de volumétrie et de stabilité de la chaîne d'exploitation.

La définition des règles de gestion a été l'un des points délicats des spécifications fonctionnelles. Quand on veut construire des séries chronologiques, il faut en effet répondre à des questions du type : « le chiffre d'affaires doit-il tenir compte des ristournes ou non ? », traiter les effets de change et les contrats dont dépendent des prix spécifiques etc.

Les référentiels

Auparavant Artichico n'avait de référentiel que pour ses produits. Les référentiels mis en place par le système d'information pour les clients (qu'il s'agisse d'individus ou d'entreprises) et pour les distributeurs ont été un atout crucial pour la réalisation du datawarehouse.

Une partie du travail était déjà faite : les identifiants des distributeurs étaient déjà définis. Pour identifier les établissements des entreprises françaises, Artichico a retenu le numéro SIRET ; pour les individus et les entreprises étrangères, l'identifiant est généré par le système.

Le traitement des changements de périmètre est assuré par des mécanismes de réaffectation de portefeuille et de reprise d'historique.

Nota Bene : On ne sait pas traiter le rattachement d'un individu à une entreprise. Il s'agit d'un lien déclaratif : comment le tenir à jour ? quels seraient les usages ? quels seraient les droits qui lui seraient attachés ? si une entreprise achète des services pour un consultant en mission auprès d'elle, à quelle entreprise rattacher cet individu ? Le rattachement d'un individu à une entreprise pose des problèmes conceptuels qui n'ont pas actuellement reçu de réponse ; il serait vain de s'acharner à l'informatiser.

Suite du projet

Après avoir achevé la mise en place du datawarehouse commercial, Artichico pourra envisager la réalisation d'un datawarehouse d'entreprise étendu aux fonctions production, maintenance, GRH, finance, stratégie. Artichico prévoit de compléter la solution actuelle par les progiciels SAS pour les analyses de données et SAP pour les analyses financières. Il sera alors possible de construire des *datamarts* spécifiques par extraction du datawarehouse d'entreprise.

8.5.2 Annexe : Les hypercubes

Un « hypercube » est, comme son nom l'indique, un tableau à n dimensions représentant la répartition d'une population selon n variables. L'œil humain ne peut lire que des tableaux à deux dimensions ; l'hypercube sera donc illisible, invisible, s'il possède plus de deux dimensions. Son utilité réside dans la multiplicité des tableaux ou séries que l'on peut obtenir, à partir d'un hypercube à n dimensions, par sommation sur $n - 2$ ou $n - 1$ indices.

Exemple : supposons que la population soit celle des clients (personnes physiques), que les critères soient la région, le mois, le métier, la tranche d'âge et le sexe. L'hypercube est alors un tableau à cinq dimensions. La case courante de l'hypercube contient le nombre de clients tel mois, dans telle région, qui avaient tel métier, tel sexe, et qui appartenaient à telle tranche d'âge. Si l'on utilise les lettres I, J, K, L et M pour désigner les indices servant à repérer les modalités de ces variables, le nombre qui figure dans la case courante de l'hypercube sera noté x_{ijklm} . Dans le langage des datawarehouses, on dit « axe » au lieu de « variable », et « segment » pour désigner l'ensemble des individus qui possèdent une même modalité de la variable.

Pour répartir une population selon les modalités d'une variable, il faut qu'il s'agisse d'une variable qualitative (comme « mois » ou « région ») ; les variables quantitatives (comme « revenu » ou « âge ») doivent, pour être représentées par un tableau, être transformées en variables qualitatives ordinales en définissant des classes (« tranche de revenu », « classe d'âge »). À partir de l'hypercube on peut, par sommation sur $n - 2$ indices, obtenir les tableaux croisant les variables deux à deux ; si l'on utilise une notation du type $\sum_m x_{ijklm} = x_{ijkl}$, la case courante du tableau qui croise les variables I et J contiendra le nombre : $\sum_{klm} x_{ijklm} = x_{ij\dots}$.

Ces tableaux peuvent être eux-mêmes redéfinis si l'on regroupe les modalités d'une variable selon une classification plus agrégée : en regroupant des mois on peut obtenir des années, en regroupant des départements on peut obtenir des régions etc.

Les logiciels usuels de datawarehouse comportent des outils commodes pour sélectionner les variables que l'on veut croiser et regrouper les modalités que l'on souhaite agréger ; des outils de représentation graphique (courbes, histogrammes, « fromages » etc.) qui facilitent la visualisation des données ; des fonctionnalités de tableur permettant de réaliser sur le tableau de nombres les calculs jugés opportuns.

L'expérience montre que les utilisateurs ont tôt fait d'apprendre à se servir de ces outils.

Chapitre 9

Modélisation par objets

« The dynamic systems created by the program execution was first and foremost a model of the system described by the program. » (Nyygard et Dahl [43] p. 483).

L'expression « technologie à objets » (on dit aussi « orientée objets ») désigne l'ensemble que constituent : (1) la « modélisation par objets », qui permet de définir les fonctionnalités qu'un système d'information doit fournir ; (2) la « programmation par objets », qui permet d'écrire les programmes du système d'information.

Dans un modèle à objets, on distingue une partie dite « métier » (ce que l'application doit faire pour répondre aux besoins des utilisateurs) et une partie technique (contraintes que l'application doit respecter pour être insérée dans le système d'information existant). La partie métier du modèle, première étape de toute réalisation, s'appuie sur la description du processus de production de valeur que l'application va équiper.

La technologie à objets, aujourd'hui bien implantée dans les entreprises, fut d'abord une innovation (Meyer [132]). Les travaux relatifs à la modélisation ont abouti à la mise au point du *langage de modélisation UML*.

9.1 Des processus aux objets

Le terme *processus* désigne l'enchaînement des tâches réalisées pour remplir une fonction de l'entreprise, c'est-à-dire pour élaborer un *produit* (voir page 157). Ces tâches sont soit mentales (perception, jugement, décision) soit physiques (fabriquer un produit, le livrer à un client, réaliser une opération de maintenance), les tâches mentales préparant et accompagnant les tâches physiques. On appelle *activité* l'ensemble des tâches réalisées par un même acteur lors d'une étape du déroulement du processus.

Le système d'information a pour fonction d'assister l'utilisateur dans les tâches mentales liées au processus : on le construira donc partir des processus, qu'il faut d'abord définir et modéliser. Cela aboutira à la réalisation d'un programme qui effectuera automatiquement les opérations suivantes :

(1) fournir à l'utilisateur les interfaces nécessaires à chaque activité (regrouper sur un écran les plages de consultation et de saisie lui évitera de se connecter, déconnecter, reconnecter à de multiples applications, de faire des doubles saisies, de naviguer dans des codes et touches de fonction divers etc.);

(2) router les messages d'une activité à la suivante (lorsque l'utilisateur tape sur la touche « valider » à la fin de son travail, il n'a pas à chercher à qui envoyer le résultat : le workflow du processus est équipé de tables d'adressage et assure automatiquement le routage);

(3) surveiller le délai de réalisation d'une activité : en cas de dépassement l'utilisateur est prévenu par une alarme, ou bien le dossier est expédié vers un autre utilisateur;

(4) produire les indicateurs (délais de réalisation, volumes traités, utilisation des ressources, satisfaction du client) qui alimenteront le pilotage du processus et permettront de vérifier la bonne utilisation des ressources humaines et matérielles.

* *

Modéliser un processus, c'est décrire la succession des activités qu'il comporte et le contenu de chaque activité : ce que fait chaque acteur, les données qu'il manipule, les traitements qu'il ordonne, les délais dans lesquels son travail doit être exécuté; c'est aussi décrire le routage des messages entre activités ainsi que les compteurs qui permettront à un animateur de contrôler la qualité du processus. La figure 9.1 représente à titre d'exemple la boucle que parcourt un processus commercial.

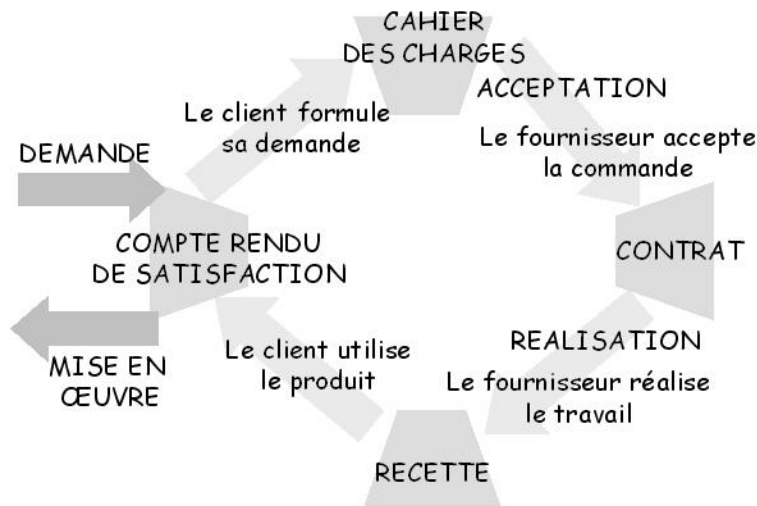


FIG. 9.1 – Boucle d'un processus commercial

La réalisation physique des tâches est décrite et balisée par le modèle du processus, mais elle nécessite des actions qui ne peuvent être réalisées que

par un être humain et qui échappent à l'ordinateur même si celui-ci aide leur préparation. Le processus relève donc de l'*assisté par ordinateur*, non de l'automatisation intégrale; il aide l'utilisateur sans se substituer à lui, tout en automatisant des tâches qu'il faisait auparavant à la main.

* *

Un processus peut se décrire sous la forme d'un graphe. Les nœuds représentent les activités, les arcs représentent le trajet des messages émis à la fin d'une activité pour lancer la (ou les) activité(s) suivante(s).

Il est préférable de donner à ce graphe une forme circulaire. Cela souligne que le processus est déclenché par un *événement* provenant de l'extérieur (réception d'une commande, d'une lettre de réclamation, franchissement du délai de maintenance d'un équipement) auquel il répond par un événement émis vers l'extérieur (livraison, lettre, opération de maintenance). Le rôle du processus, c'est de réaliser l'ensemble des tâches qui concourent à l'élaboration de cette réponse et qui constituent l'acte de production. Il convient de s'assurer que la réponse est émise dans un délai et sous la forme convenable, qu'elle est de bonne qualité : c'est le *contrôle du bouclage* (figure 9.2).

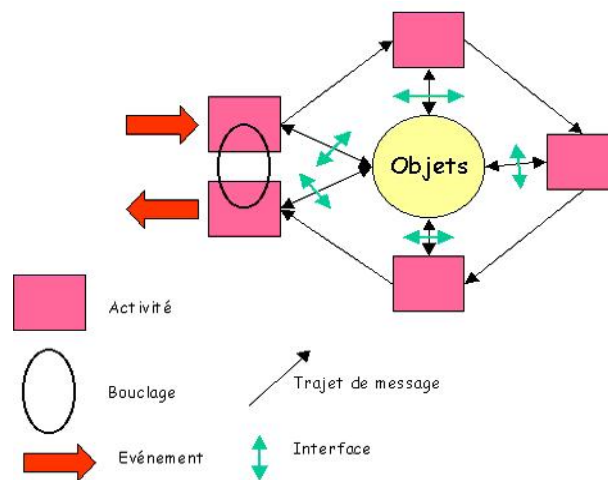


FIG. 9.2 – Bouclage d'un processus

Les activités qui s'enchaînent lors d'un processus comportent des saisies de données, des consultations de données, le lancement de traitements sur les données. Les données consultées et saisies lors du processus doivent être cohérentes : s'il s'agit par exemple de traiter une commande, il faudra que les ordres de production ou de déstockage reprennent fidèlement ses termes. Ainsi les activités « font la ronde autour des données » ; nous avons représenté dans la figure 9.2 les données (et les traitements qui leur sont associées) par un rond marqué du mot « Objets », car l'une des façons les plus efficaces de représenter données et traitements est de les organiser en « objets ».

Le rôle du processus n'est pas de répondre à un événement isolé, mais à un flux d'événements de même nature : le processus de production répond à un flux de commandes ; le processus de réponse au courrier répond à un flux de lettres. Le caractère répétitif et circulaire du processus est analogue à celui d'un moteur même si son rythme, dicté par l'arrivée aléatoire des événements, n'a pas la même régularité. La qualité du processus s'évalue non sur un événement particulier, mais sur une statistique (histogramme de délais, de satisfaction des clients etc.) représentant la façon dont l'ensemble des événements a été traité.

Le respect du délai de réponse implique souvent une délégation de responsabilité aux personnes qui réalisent les tâches. L'approche par les processus suscite donc une modification du rôle de l'encadrement : son intervention ne se fonde plus sur l'approbation des actes un par un, mais sur un contrôle statistique *a posteriori* et sur la mise à jour des consignes si un dysfonctionnement apparaît ou s'il faut faire évoluer le processus. La transmission des consignes vers les exécutants, l'émission des rapports vers les responsables sont remplacées en partie, l'une par la mise en forme du workflow, l'autre par l'édition semi-automatique de comptes rendus alimentés par les compteurs que le processus comporte. Le nombre des niveaux hiérarchiques se réduit alors et la communication entre « base » et « sommet » devient moins lointaine.

La réalisation d'un processus suppose des sous-processus fournissant chacun des produits intermédiaires ou « livrables » (exemple : expertise fournie lors de l'instruction d'une demande d'autorisation d'investissement ou d'une demande de crédit). En pratique, les sous-processus sont nombreux et comportent eux-mêmes des sous processus ; la figure 9.3 est donc schématique.

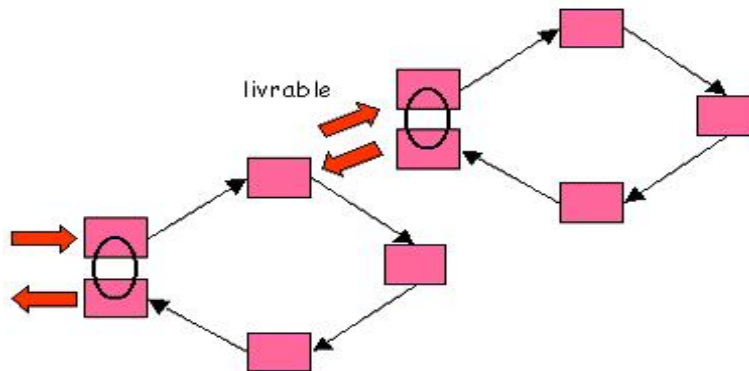


FIG. 9.3 – Sous-processus et livrables

Toute opération de production est une réponse à un événement extérieur et articule des activités diverses : le processus existe donc *de facto* avant toute description, toute modélisation, de même qu'une langue existe avant que l'on n'ait rédigé sa grammaire. Mais un processus qui n'a pas été pensé présente souvent des défauts. Par exemple la succession des tâches se poursuivra sans que l'on vérifie s'il a été répondu à chaque événement ; le

risque existe alors que le processus ne « se perde dans les sables ». C'est le cas lorsque la lettre d'un client passe de bureau en bureau, et attend dans des piles sans que personne ne contrôle le délai de réponse : on renoncera à répondre lorsque le délai décent aura été dépassé et la lettre ira à la corbeille. Le dernier document arrivé sur un bureau est souvent traité en premier, ce qui accroît la dispersion des délais de traitement.

La modélisation d'un processus n'est pas la simple mise en forme du processus existant, car elle conduit souvent à repérer et corriger des défauts : bras morts où les délais s'accumulent ; redondance (la même tâche est réalisée par deux acteurs à la fois) ; erreur d'adressage (le dossier parvient à une personne non concernée qui devra identifier le destinataire et le lui faire parvenir) ; mauvaise répartition de la charge de travail (certaines personnes sont surchargées alors que d'autres n'ont rien à faire) etc. Comme l'entreprise travaille selon des habitudes dont la raison d'être s'est souvent estompée, un processus qui n'a jamais été modélisé présente presque toujours des défauts : la modélisation fait souvent gagner de l'ordre de 20 % en productivité comme en qualité.

On dit parfois que la modélisation doit « optimiser » le processus. L'expérience montre que le gain de qualité provient de la clarté qu'apporte la modélisation, de l'animation qu'elle permet de mettre en place, plutôt que d'un effort tendu vers la recherche d'un optimum qu'il serait d'ailleurs difficile de définir.

* *

Chaque utilisateur va, lors de l'activité qu'il accomplit, consulter ou saisir quelques données, déclencher quelques traitements : ceci conduit naturellement vers la programmation par objets. Le mot « objet » désigne ici un petit programme qui rassemble les données et les traitements concernant un des êtres du monde réel sur lesquels l'entreprise entend agir (produit, client, fournisseur, partenaire etc.), ou un être intermédiaire (dossier, rapport, feuille de calcul) nécessaire à l'élaboration du produit.

Pour modéliser le processus de production, on dispose de plusieurs langages : UML (Unified Modeling Language) convient bien à la préparation d'une programmation par objets. Pour les petits processus¹, qui peuvent être réalisés avec des outils bureautiques (demandes de congé ou de mutation, demandes de fournitures, préparation du budget annuel etc.), on peut se contenter des outils de workflow offerts sur le marché.

9.2 De la programmation impérative à l'objet

Tout programme consiste en la définition :

(1) des structures de données qui seront manipulées (saisies, consultées, soumises à des traitements) ;

1. Si ces processus sont « petits » selon le volume des ressources informatiques consommées et le coût de leur mise en place, ils peuvent être importants pour l'efficacité de l'entreprise : il ne faut donc pas les négliger.

- (2) des algorithmes qui seront appliqués aux données ;
- (3) d'une succession (comportant éventuellement des boucles, fourches et chemins parallèles) d'appels de données et de traitements.

Les langages de programmation se distinguent par la façon dont ils réalisent et articulent ces trois éléments, notamment par les règles de nommage et d'adressage qui permettent de localiser et d'appeler données et traitements.

Selon la thèse de Church-Turing, tout langage de programmation non trivial équivaut à une machine de Turing ; il en résulte que tout programme qu'il est possible d'écrire dans un langage pourrait également être écrit dans n'importe quel autre langage. La différence entre les langages ne réside donc pas dans ce qu'ils permettent ou non de programmer, mais dans ce qu'il est facile, commode de programmer avec chaque langage : cette différence est d'ordre non pas logique, mais pratique.

Elle n'en est pas moins importante et il faut relativiser la thèse de Church-Turing. Utilisons une analogie : il est certes également possible de traverser la Seine en marchant sur un câble d'acier ou sur le pont de la Concorde ; mais pour marcher sur le premier il faut être un funambule bien entraîné alors que passer par le second est à la portée d'un promeneur.

Il en est de même pour les langages. Tout ce que l'on fait avec un langage de programmation par objets pourrait être fait en programmation impérative ; mais la sécurité n'est pas la même dans les deux cas, ni la possibilité de faire évoluer le programme pour un coût raisonnable.

9.2.1 Programmation impérative

Si l'on peut tout faire avec un langage de programmation, et notamment faire en sorte qu'une application respecte toutes les contraintes de qualité évoquées page 286, ce sera plus ou moins difficile selon la démarche de modélisation et le langage de programmation que l'on utilise (la « modélisation », c'est la définition des structures de données et des traitements ; elle est logiquement et chronologiquement antérieure à la programmation proprement dite). Il est trop facile de réaliser des « plats de spaghetti » en Fortran, en Cobol ou en C ; Pascal incite à la modularité, ce qui encourage à rédiger des programmes clairs et bien documentés, mais comme il se prête mal à l'industrialisation il est peu utilisé par les programmeurs professionnels.

Ces quatre langages sont destinés à la programmation impérative, celle qui correspond à l'intuition la plus immédiate pour un informaticien : il s'agit de programmer d'une part les structures de données, d'autre part les traitements qui transformeront les données d'entrées en résultats. Elle se prête à la mise en commun des traitements (le traitement « mise à jour » peut être utilisé par les fonctions « créer » et « supprimer ») comme à la mise en commun des données qui seront éventuellement partagées avec d'autres applications (elles sont alors classées dans une « base de données »).

Quoique données et traitements soient solidaires, puisque le logiciel fait traiter les données par l'ordinateur, l'approche la plus courante a été de concevoir les bases de données séparément des traitements et d'utiliser pour

les construire des outils de modélisation et de documentation différents. Certes il est avantageux de traiter séparément des problèmes de nature différente, mais cela présente un inconvénient : si l'application est compliquée et si les besoins évoluent (création de nouveaux produits, introduction de nouveaux segments de clientèle, de nouveaux types de partenaires etc.), les mises à jour obligeront à réécrire une grande partie du code au risque d'introduire des erreurs qu'il sera difficile de repérer et de corriger.

Rien n'empêche d'organiser la programmation impérative de sorte que les évolutions soient aussi simples que possible : cela suppose, en pratique, de rapprocher données et traitements. Si par exemple une entreprise a défini la structure de données et les traitements pour traiter des abonnés à un service, et si elle veut par la suite que la même application puisse traiter aussi des acheteurs de matériel, il lui faudra introduire une nouvelle structure de données et de nouveaux traitements et cela l'obligera à réécrire une partie de l'application. Mais si elle a pris la précaution de définir une structure de données « client » susceptible de comporter comme cas particuliers des abonnés, des acheteurs de matériels ou d'autres types de clients, elle pourra introduire un nouveau type de client sans écrire autre chose que le code strictement nécessaire. Cependant, comme certains traitements s'appliquent à l'abonné et d'autres à l'acheteur, il faudra encore que les traitements associés à la structure de données « client » soient paramétrés par le type de client : il sera donc utile, pour la clarté du code, d'écrire tout près l'un de l'autre les données et traitements relatifs à la structure « client ». Nous sommes dès lors proches de la définition de ce que les informaticiens appellent un objet.

Ce petit exemple montre comment les informaticiens ont très naturellement été conduits, pour améliorer l'évolutivité des applications, à concevoir des langages dans lesquels les données qui décrivent un dossier sont insérées dans le même module que les traitements qui leur sont appliqués ; pour utiliser les termes consacrés, on dira alors que données et traitements sont « encapsulés dans le même objet ».

Ainsi les langages de programmation par objets (ou « orientée objets ») sont la conséquence ultime de la modularisation du logiciel, démarche qui vise à maîtriser sa production et son évolution. Cependant, et malgré cette continuité logique, ces langages ont apporté en pratique un profond changement dans l'art de la programmation : même si la programmation par objets n'est que de la programmation impérative bien organisée, elle implique en effet un bouleversement de l'attitude mentale du programmeur.

En programmation impérative, la qualité des méthodes devait être apportée par le programmeur lui-même, un programmeur peu rigoureux ayant toute liberté de programmer un « plat de spaghetti » peu évolutif. En programmation par objets, une part de la rigueur est incorporée dans le langage. Cela n'exclut pas tout risque d'erreur : même si un pont est muni de garde-fous, quelqu'un qui voudrait se jeter dans la rivière, tourner indéfiniment en rond sur le pont ou le franchir en marchant sur les mains pourrait le faire ; mais l'utilisateur de bon sens, qui veut tout simplement traverser le pont, le traversera facilement en marchant selon son ordinaire.

9.2.2 Programmation par objets

Le mot « objet » est un faux ami : il oriente dans une mauvaise direction l'intuition de celui qui l'entend pour la première fois et il faut lutter contre cette intuition pour comprendre de quoi il s'agit (voir page 35). Pour faciliter la compréhension, nous allons recourir à des analogies.

Toute entreprise doit gérer des « individus » qui composent des « populations », en utilisant ces termes selon le sens qu'ils ont en statistique : les « individus » peuvent être des personnes physiques ou morales, des pièces de rechange, machines, établissements, commandes, factures etc. (« individu » est alors synonyme d'« existant »). Une « population » est un ensemble fini d'individus que l'on a regroupés parce qu'ils possèdent des caractéristiques jugées analogues.

Dès que les entreprises se sont organisées elles ont construit des « dossiers » pour décrire les individus (produits, entités de l'organisation, clients etc.) ; à chaque population correspondait un type de dossier (ou de formulaire, ou de questionnaire) définissant une liste de données, à chaque individu correspondait un dossier rempli, chaque case contenant la valeur d'une donnée observée sur l'individu. À chaque dossier étaient associés également quelques calculs : vérifications et recoupements pour s'assurer de l'exactitude des données ; évaluation des taxes, droits, montants de facture etc.

La modélisation par objets, même si elle semble nouvelle, ne fait que revêtir d'un vocabulaire nouveau cette démarche ancienne. La « classe », description de l'individu type d'une population, n'est rien d'autre qu'un dessin de dossier dans lequel on aurait fait figurer, outre la définition des données (que l'on appelle « attributs »), celle des traitements qui leur sont associés (que l'on appelle « opérations » ou « méthodes ») ; et le dossier relatif à un individu, dans lequel sont inscrites (« instanciées ») les valeurs des données observées sur cet individu, sera nommé « objet ».

Ce vocabulaire provoque des confusions. La « classe » du langage à objets n'est pas la « classe » d'une classification (qui est, elle, l'un des éléments d'une partition opérée sur une population) ; l'« objet » du langage à objets, qui représente un individu, n'est pas l'« objet » du langage courant, qui est l'individu lui-même ; les « méthodes » du langage à objets ne sont pas des méthodes au sens où l'on dit « méthode de travail », mais des fonctions au sens mathématique du terme comme lorsqu'on écrit $y = f(x)$.

Lorsqu'un informaticien dit « un objet est une abstraction », cette phrase est exacte car il parle d'un objet au sens de la modélisation par objets : en effet la liste des données que contient cet objet, étant sélective, est « abstraite » de l'ensemble infini des données que l'on pourrait observer sur un même individu. Mais comme le philosophe entend par « objet » l'individu lui-même, cette phrase est pour lui un non-sens. Cependant il aurait tort de se laisser arrêter par cette contrariété superficielle : s'il accepte de comprendre le vocabulaire de la modélisation par objets, il verra que celui-ci ne fait qu'organiser de façon ingénieuse la démarche d'abstraction et de conceptualisation qui, depuis Aristote, se trouve au fondement même de la philosophie.

Le modélisateur choisit la liste des attributs à retenir pour caractériser une classe. Ce choix est guidé par les exigences de pertinence et de sobriété du système d'information : il faut connaître les données utiles à l'action que l'on entend assister, et elles seules.

Supposons que pour une application particulière on ait défini la classe `automobile` et retenu la liste suivante des attributs et des méthodes (figure 9.4).

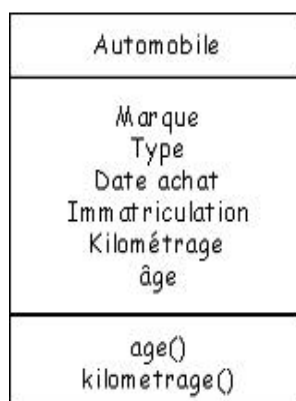


FIG. 9.4 – Représentation d'une classe

Il est d'usage de terminer le nom d'une méthode par deux parenthèses successives. Ici, nous supposons qu'`age()` calcule l'âge d'une automobile par différence entre la date d'achat et la date du jour et que `kilometrage()` assure la mise à jour du kilométrage cumulé après la saisie du kilométrage hebdomadaire.

À chaque automobile sera associé un « objet » où figurent les instances des divers attributs. Chaque objet comporte (1) un identifiant permettant de désigner sans ambiguïté l'individu dont il s'agit (ici : numéro du châssis) ; (2) les valeurs des attributs observées sur cet individu ; (3) les méthodes définies pour les individus de la classe (figure 9.5).

Les attributs qui caractérisent un objet sont protégés contre des modifications intempestives : ils ne peuvent être modifiés que si l'utilisateur emprunte la procédure prévue, que l'on appelle « interface ». La protection ainsi accordée aux données est nommée « encapsulation ». C'est l'un des garde-fous que procure la programmation par objets : en programmation impérative, si l'on n'y prend garde, les données pourraient être modifiées sans précaution particulière.

* *

La modélisation par objets utilise des procédés qui permettent d'économiser l'écriture du code et contribuent à son évolutivité :

Certaines populations peuvent faire l'objet d'une segmentation en sous-populations : c'est le cas des classes « employés », « client », « produit »,

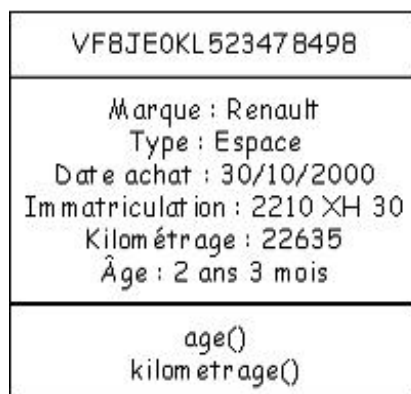


FIG. 9.5 – Représentation d'un objet

« fournisseur » etc. Chaque sous-population peut être caractérisée par les mêmes données que la population entière, auxquelles on ajoute des données propres à cette sous-population. Si par exemple on considère la population des « employés », on peut définir les sous-populations « *managers* », « ouvriers », « secrétaires » etc. caractérisées chacune par des données spécifiques venant s'ajouter aux données qui caractérisent les « employés » en général.

Dans le langage de la modélisation par objets, on dira que les classes « manager », « ouvrier », « secrétaire » *héritent* de la classe « employé », car elles reprennent ses attributs et méthodes en leur ajoutant des attributs et méthodes spécifiques.

L'héritage peut être partiel : dans ce cas, seule une partie des attributs ou méthode de la classe-mère est repris par la classe-fille. Il peut être simple ou multiple : une même classe peut (mais c'est une possibilité que l'on utilise rarement) être sous-classe de plusieurs classes. Toute modification de la classe-mère entraîne une modification des classes-filles sans qu'il soit nécessaire de reprogrammer celles-ci. En sens inverse, il arrive que plusieurs classes puissent être considérées comme filles d'une même classe-mère, ce qui permet de regrouper leurs caractéristiques communes en créant cette classe-mère.

Le programmeur peut construire rapidement les classes dont il a besoin en puisant dans le stock des classes existantes, les « bibliothèques ». Les bibliothèques fournies avec un langage à objets sont riches ; le premier travail du programmeur sera de sélectionner et charger les classes utiles, après quoi la programmation ressemblera au montage d'un produit vendu en kit.

Le programmeur peut bien sûr constituer une bibliothèque avec les classes qu'il a définies lui-même, ce qui lui permettra de réutiliser son propre code (*Nota Bene* : cet argument est souvent évoqué en faveur des langages objet ; toutefois la réutilisation suppose chez le programmeur une rigueur peu fréquente).

Le *polymorphisme* est, pour une même opération, la faculté de s'appliquer à des objets appartenant à des classes différentes.

Exemple : pour calculer la surface d'un carré, on utilise la formule $S = a^2$; pour un triangle, $S = Ah/2$; pour un cercle, $S = \pi R^2$; pour un trapèze, $S = (A+B)h/2$ etc. Ainsi la même opération « calculer la surface » s'appuie, selon la figure géométrique considérée, sur des méthodes différentes.

Si l'opération `calculerlasurface()` est douée de polymorphisme, elle utilisera la méthode adaptée quelle que soit la figure géométrique concernée. Dès lors le programmeur n'a plus à se soucier du type de figure auquel cette opération s'appliquera.

L'*agrégation* indique qu'une classe est une composante d'une autre classe. Cela permet de définir des objets composés d'autres objets, que l'on nomme « composants » ou « packages ». L'une des classes agrégées dans un composant est la classe principale (*main*) qui assure les fonctions d'interface avec le reste du monde et lance les traitements dans les autres classes.

Le composant « client » peut par exemple contenir une classe « adresse » qui permet de noter la (ou les) adresses du client. Cette même classe « adresse » pourra être utilisée par le composant « fournisseur », le formalisme de l'adresse ne changeant pas selon que l'on considère l'un ou l'autre des composants.

* *

Avant d'écrire la première ligne d'un programme à objets, il importe d'avoir défini les classes, attributs etc., c'est-à-dire d'avoir « modélisé » l'application. Ce travail sera fait soit par le programmeur, qui devra faire valider ses choix par le client, soit par la maîtrise d'ouvrage qui livrera au programmeur un « modèle métier » (ou « spécifications générales ») proche des préoccupations du client mais déjà formulé selon les concepts de la modélisation par objets (classes, attributs etc.). L'état de l'art propose d'utiliser, pour cette modélisation, le langage UML. Ce modèle sera progressivement précisé jusqu'à ce que toutes les décisions nécessaires à la production du code aient été prises.

Le programmeur lance alors, en s'appuyant sur le modèle, un générateur de code qui produira automatiquement de l'ordre de 80 % des lignes de code ; il restera à les compléter pour finir l'écriture du programme. Un outil comme Rose de Rational (www.rational.com/products/rose/index.jsp) permet à la fois d'écrire le modèle, dans ses diverses étapes, et de générer une partie du code : ainsi le travail peut progresser de façon cumulative, sans perte d'information et sans qu'il soit besoin de refaire le travail déjà fait pour l'adapter à un formalisme nouveau.

La programmation par objets requiert une réflexion préalable plus longue que celle, peut-être insuffisante, qu'il était habituel de consacrer à la programmation impérative. Par contre une fois le modèle construit l'écriture du code est plus rapide.

Lorsqu'un programme à objets est lancé, l'exécution commence par une des classes (nommée « main », classe principale). Cette classe envoie des messages vers d'autres classes ou sollicite une action de l'opérateur humain (ou

d'une autre application), ce qui déclenche une cascade de traitements, messages et affichages sur les équipements périphériques (écrans, imprimantes etc.) jusqu'à la fin de l'exécution du programme.

* *

Les objets communiquent par des messages qui transmettent des données ou des appels de méthodes, les adresses des objets destinataires étant définies dans l'espace de nommage du programme. L'adressage et le libellé des messages posent problème lorsqu'on travaille sur des objets résidant sur des ordinateurs différents (« objets distribués »), appartenant à des programmes écrits dans des langages différents ou exploités sur des plates-formes différentes. Il faut alors utiliser un « Object Request Broker » (ORB) : cela permet de concevoir des applications où le traitement réalisé sur un ordinateur nécessite l'exécution d'un autre traitement sur un autre ordinateur. Les objets distribués sur la Toile sont nommés « Web services ».

La définition des ORB a nécessité une normalisation qui a été assurée par l'OMG (Object Management Group), www.omg.org, organisation internationale qui fédère les entreprises concernées par la technologie objets). L'OMG a défini l'architecture MDA (« Model Driven Architecture »), dont les fleurons sont la norme CORBA (« Common Object Request Broker Architecture ») pour les ORB et le langage UML (« Unified Modeling Language ») pour la modélisation. Les outils CORBA relèvent, comme les EAI, du « middleware » : ils assurent des fonctions de communication entre programmes et remplissent donc *mutatis mutandis* une fonction analogue à celle des commutateurs d'un réseau téléphonique ou des « hubs » d'un transporteur aérien.

* *

Pour longtemps encore le système d'information d'une entreprise devra faire cohabiter les applications nouvelles, écrites en langage à objets, et les applications anciennes qui gèrent séparément les bases de données et les traitements. Il est possible d'introduire dans un programme à objets des êtres qui se comportent comme des objets quand on les voit du côté du langage à objets, et qui par ailleurs lancent des requêtes et appels de traitements vers l'application traditionnelle.

Sun commercialise à cette fin la plate-forme J2EE (« Java 2 Platform, Enterprise Edition ») qui assure la communication entre les applications objet, les bases de données et les applications traditionnelles.

La mise en œuvre des brokers, Web Services et interfaces diverses est, en pratique, des plus délicates. Les langages de haut niveau dans lesquels les programmes sont écrits masquent en effet ce qui se passe réellement et physiquement dans les compilateurs, processeurs et mémoires, et le programmeur peut avoir de mauvaises surprises (le programme est conforme aux bonnes règles d'écriture, mais il donne pourtant un résultat faux). Par ailleurs la cascade des transcodages, traductions, interprétations, appels de traitement, requêtes etc. peut aboutir, en termes de performances, à des délais que l'utilisateur juge intolérables. Ces problèmes-là relèvent de l'art de l'informaticien expert.

9.2.3 Origines de la programmation par objets

Les premiers langages de programmation qui aient utilisé des objets sont Simula I (1961-64) et Simula 67 (1967), conçus par les informaticiens norvégiens Ole-Johan Dahl et Kristan Nygaard pour traiter numériquement des problèmes qu'il serait impossible de résoudre de façon analytique.

Simula a ainsi été utilisé pour calculer la distribution statistique des durées de traitement des passagers dans un aéroport, problème où s'articulent plusieurs phénomènes aléatoires (rythme d'arrivée des passagers, durée de l'enregistrement, durée des déplacements dans l'aéroport etc.). Simula représentait chaque passager par un objet (nombre de bagages, destination etc.), et après un nombre suffisant de simulations fournissait les histogrammes des distributions statistiques recherchées. Simula 67 contenait déjà les objets, les classes, l'héritage, l'encapsulation etc. (Nygaard et Dahl [43]).

Alan Kay, du PARC de Xerox, avait utilisé Simula dans les années 1960. Il réalisa en 1976 Smalltalk qui reste aux yeux de certains programmeurs le meilleur langage de programmation par objets. Pour Alan Kay, chaque objet est comme un petit ordinateur qui interagirait avec d'autres ordinateurs : « Bob Barton avait dit : "le principe de base de la récursion est de faire en sorte que les parties soient aussi puissantes que le tout". Pour la première fois, j'envisageai l'ordinateur comme le tout ; je me suis alors demandé pourquoi on le divisait en entités plus faibles que lui, comme les structures de données et les procédures. Pourquoi ne pas le diviser en petits ordinateurs comme commençait à le faire le temps partagé ? Et pourquoi pas des milliers d'ordinateurs, et non seulement des douzaines, qui simuleraient chacun une structure utile ?² »

Bjarne Stroustrup a mis au point C++ aux Bell Labs d'AT&T en 1980. C++, c'est « C with classes », un langage C doté des outils qui permettent la programmation par objets. C++ deviendra le langage le plus utilisé par les programmeurs professionnels ; il requiert beaucoup de savoir faire, car comme C il permet d'agir sur les couches basses (programmer un pilote de disque ou une pile TCP/IP), possibilité qui est utile pour un industriel mais dangereuse entre les mains d'un débutant et qui ne sera d'ailleurs généralement pas nécessaire s'il s'agit de programmer une application destinée à une entreprise.

Java est lancé par Sun en 1995. Comme il présente plus de sécurité que C++ il deviendra le langage favori de certains programmeurs professionnels. Un programme en Java peut être soit directement compilé en langage machine (tout comme un programme en C++), soit d'abord compilé dans un code intermédiaire, le « bytecode », puis interprété et exécuté sur chaque machine par une « Java Virtual Machine ». Ce dispositif en deux étapes permet d'exécuter un même programme Java sur des plates-formes diverses,

2. « Bob Barton (...) had said (...): "The basic principle of recursive design is to make the parts have the same power as the whole". For the first time I thought of the whole as the entire computer and wondered why anyone would want to divide it up into weaker things called data structures and procedures. Why not divide it up into little computers, as time-sharing was starting to? But not in dozens. Why not thousands of them, each simulating a useful structure? » (Alan Kay, [98] p. 516).

pourvu qu'elles soient munies chacune de sa JVM. Cela permet à Java de réaliser l'un des objectifs les plus ambitieux en informatique : « write once - run anywhere », un programme une fois écrit peut être exploité sur toutes les plates-formes. En outre Java fournit au programmeur des outils commodes comme celui qui libère automatiquement la mémoire non utilisée (« garbage collector »). Cette fonction faisait défaut en C et C++, où la gestion de la mémoire est un casse-tête pour le programmeur.

On doit ajouter à cette liste plusieurs autres langages : Eiffel, Objective C, Loops etc. Presque tous les langages impératifs se sont dotés d'outils de programmation par objets. Les programmeurs compétents en programmation par objets sont les plus demandés sur le marché. Les langages de programmation par objets sont entourés d'outils de productivité pour le programmeur professionnel : générateurs automatiques de code, bibliothèques de classes, environnement de développement intégré, outils de test etc.

Faut-il désormais tout programmer en langage à objets ? Pas nécessairement. Pour des applications simples, ou entre les mains d'un programmeur rigoureux, la programmation impérative est efficace - et on peut faire beaucoup de choses en utilisant tout simplement des logiciels bureautiques comme Excel, Word, PowerPoint, Access et FrontPage ! Par ailleurs, l'essentiel des applications exploitées dans les entreprises a été programmé voici longtemps en Cobol ; elles fonctionnent correctement et il faudra pendant des années encore assurer leur maintenance, ce qui assure une longue vie à la programmation en Cobol.

9.3 Langage de modélisation UML

Références sélectives

Sur la Toile : www.uml.org et uml.free.fr (cours en français)

Bibliographie :

Grady Booch, Ivar Jacobson, James Rumbaugh, *The Unified Modeling Language User Guide* [19]

Martin Fowler et Kendall Scott, *UML distilled second edition* [181]

Pascal Roques et Franck Vallée, *UML en action* [207]

La description de la programmation par objets a fait ressortir l'étendue du travail conceptuel nécessaire : définition des classes, de leurs relations, des attributs et méthodes, des interfaces etc. Si l'on a compris l'analogie entre « classe » et « type de dossier », entre « objet » et « dossier individuel rempli » etc., on voit que l'énoncé des choix ci-dessus n'est rien d'autre que la modélisation, ou « spécification », du programme.

Pour programmer une application, il ne convient pas de se lancer tête baissée dans l'écriture du code : il faut d'abord organiser ses idées, les documenter, puis organiser la réalisation en définissant les modules et étapes de

la réalisation. C'est cette démarche antérieure à l'écriture que l'on appelle « modélisation » ; son produit est un « modèle ».

Les spécifications fournies par la maîtrise d'ouvrage en programmation impérative étaient souvent floues : les articulations conceptuelles (structures de données, algorithmes de traitement) s'exprimant dans le vocabulaire de l'informatique, le modèle devait souvent être élaboré par celle-ci. L'approche objet permet en principe à la maîtrise d'ouvrage de s'exprimer de façon précise selon un vocabulaire qui, tout en transcrivant les besoins du métier, pourra être immédiatement compris par les informaticiens. Nous avons dit *en principe* parce que la modélisation demande aux maîtrises d'ouvrage une compétence, un professionnalisme qui ne sont pas aujourd'hui répandus.

9.3.1 Histoire des modélisations par objets

Les méthodes utilisées dans les années 1980 pour organiser la programmation impérative (notamment Merise) étaient fondées sur la modélisation séparée des données et des traitements.

Lorsque la programmation par objets prend de l'importance au début des années 1990, la nécessité d'une méthode qui lui soit adaptée devient évidente. Plus de cinquante méthodes apparaissent entre 1990 et 1995 (Booch, Classe-Relation, Fusion, HOOD, OMT, OOA, OOD, OOM, OOSE etc.) mais aucune ne parvient à s'imposer. En 1994, le consensus se fait autour de trois méthodes :

- OMT de James Rumbaugh (General Electric) fournit une représentation graphique des aspects statique, dynamique et fonctionnel d'un système ;
- OOD de Grady Booch, définie pour le Department of Defense, introduit le concept de *package* ;
- OOSE d'Ivar Jacobson (Ericsson) fonde l'analyse sur la description des besoins des utilisateurs (cas d'utilisation, ou *use cases*).

Chaque méthode avait ses avantages et ses partisans. Le nombre de méthodes en compétition s'était réduit mais le risque d'un éclatement subsistait : la profession pouvait se diviser entre ces trois méthodes, créant autant de continents intellectuels qui auraient du mal à communiquer.

Événement considérable et presque miraculeux, les trois gourous qui régnaient chacun sur l'une des trois méthodes se mirent d'accord pour définir une méthode commune qui fédérerait leurs apports respectifs (on les surnomme depuis « the Amigos »). UML, « Unified Modeling Language », est né de cet effort de convergence. L'adjectif « unified » est là pour marquer qu'UML « unifie », et donc remplace, les méthodes antérieures (voir Cris Kobryn, « UML 2001: A Standardization Odyssey », *Communications of the ACM*, octobre 1999.). En fait, et comme son nom l'indique, UML n'a pas l'ambition d'être exactement une méthode : c'est un *langage*.

L'unification a progressé par étapes. En 1995, Booch et Rumbaugh (et quelques autres) se sont mis d'accord pour construire une méthode unifiée, « Unified Method 0.8 » ; en 1996, Jacobson les a rejoints pour produire UML 0.9 (noter le remplacement du mot « méthode » par le mot « langage », plus modeste). Les acteurs les plus importants dans le monde du logiciel

s'associent alors à l'effort (IBM, Microsoft, Oracle, DEC, HP, Rational, Unisys etc.) et UML 1.0 est soumis à l'OMG. L'OMG adopte en novembre 1997 UML 1.1 comme langage de modélisation des systèmes d'information à objets. La version d'UML en cours à la fin 2002 est UML 2.0 et les travaux d'amélioration se poursuivent.

UML est donc non seulement un outil intéressant mais une *norme* qui s'impose en technologie à objets et à laquelle se sont rangés tous les grands acteurs du domaine (ils ont d'ailleurs contribué à son élaboration). Chacun est libre de critiquer UML (nous formulerons d'ailleurs quelques critiques) mais il faut respecter le résultat d'un effort de normalisation dans la modélisation, domaine si difficile à formaliser.

9.3.2 UML en œuvre

UML n'est pas une *méthode*, une description normative des étapes de la modélisation : ses auteurs ont en effet estimé qu'il n'était pas opportun de définir une méthode en raison de la diversité des cas particuliers. Ils ont préféré se borner à définir un *langage graphique* qui permet de représenter, de communiquer les divers aspects d'un système d'information (aux graphiques sont bien sûr associés des textes qui expliquent leur contenu). UML est donc un *métalangage* car il fournit les éléments permettant de construire le modèle qui, lui, sera le langage de l'entreprise.

Un système d'information est un être organique : il articule plusieurs logiques qui jouent simultanément et que l'on peut représenter par un modèle en couches. Il est impossible de donner une représentation graphique complète d'un être organique, de même qu'il est impossible de représenter entièrement une statue (à trois dimensions) par des photographies (à deux dimensions). Mais il est possible de donner sur un tel être des « vues » partielles, analogues chacune à une photographie d'une statue, et dont la juxtaposition donnera une idée utilisable en pratique sans risque d'erreur grave. UML comporte ainsi douze diagrammes standard représentant autant de « vues » du système d'information. Ils se répartissent en trois catégories : quatre représentent la structure statique de l'application (diagrammes de classe, d'objet, de composant et de déploiement) ; cinq représentent son comportement dynamique (diagrammes de cas d'utilisation, de séquence, d'activité, de collaboration et d'état) ; trois représentent la façon dont on peut organiser et gérer les modules qui composent le programme (diagrammes de packages, sous-systèmes et modèles).

Ces diagrammes, d'une utilité variable selon les cas, ne sont pas nécessairement tous produits à l'occasion d'une modélisation. Les plus utiles pour la maîtrise d'ouvrage sont les diagrammes d'activité, de cas d'utilisation, de classe, d'objet, de séquence et d'état. Les diagrammes de composants, de déploiement et de collaboration sont surtout utiles pour la maîtrise d'œuvre à qui ils permettent de formaliser les contraintes de la réalisation et la solution technique³.

3. Les diagrammes fournis en exemple ont été aimablement communiqués par M. Mat-

Le *diagramme d'activité* n'est autre que la transcription dans UML de la représentation du processus telle qu'elle a été élaborée lors du travail qui a préparé la modélisation (voir page 436) : il montre l'enchaînement des activités qui concourent au processus.

Le *diagramme de cas d'utilisation* (figure 9.6) décrit la succession des opérations réalisées par un acteur (personne qui assure l'exécution d'une activité). C'est le diagramme principal du modèle UML, celui où s'assure la relation entre l'utilisateur et les objets que le système met en œuvre.

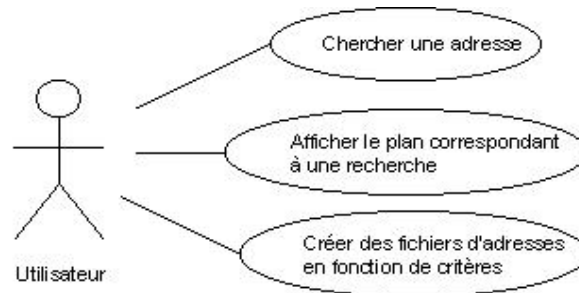


FIG. 9.6 – Diagramme de cas d'utilisation

Le *diagramme de classe* (figure 9.7) représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens, que ceux-ci représentent un emboîtement conceptuel (héritage, marqué par une flèche terminée par un triangle) ou une relation organique (agrégation, marquée par une flèche terminée par un losange ou « diamant »).

Le *diagramme d'objet* permet d'éclairer un diagramme de classe en l'illustrant par des exemples.

Le *diagramme de séquence* (figure 9.8) représente la succession chronologique des opérations réalisées par un acteur : saisir une donnée, consulter une donnée, lancer un traitement ; il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre. On peut représenter les mêmes opérations par un *diagramme de collaboration*, graphe dont les nœuds sont des objets et les arcs (numérotés selon la chronologie) les échanges entre objets : diagramme de séquence et diagramme de collaboration sont deux vues différentes mais logiquement équivalentes (on peut construire l'une à partir de l'autre) d'une même chronologie.

Le *diagramme d'état* représente la façon dont évoluent (« cycle de vie ») durant le processus les objets appartenant à une même classe. La modélisation du cycle de vie est essentielle pour représenter et mettre en forme la dynamique du système.

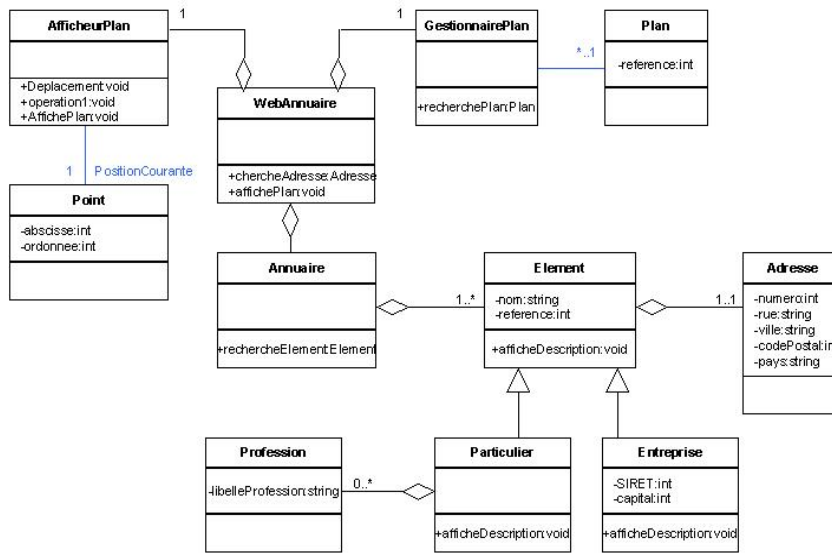


FIG. 9.7 – Diagramme de classe

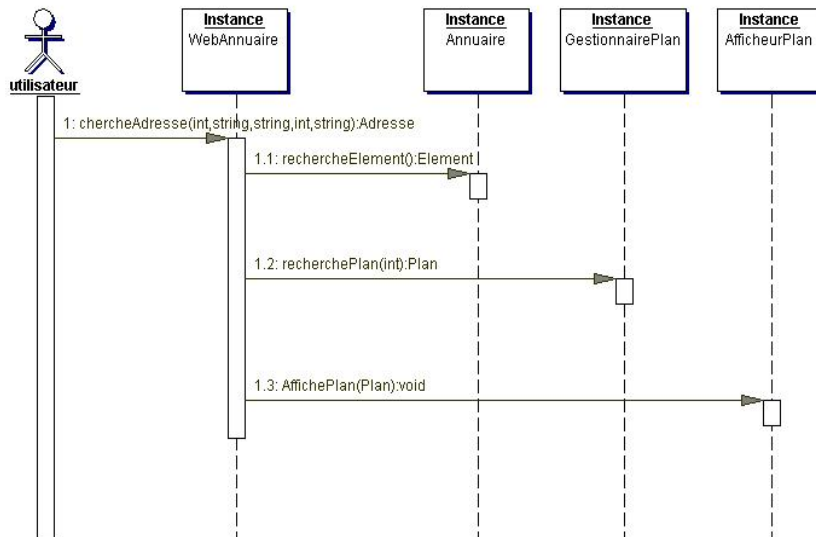


FIG. 9.8 – Diagramme de séquence

9.3.3 Comment présenter un modèle UML ?

La présentation d'un modèle UML se compose de plusieurs documents écrits en langage courant et d'un document formalisé (figure 9.9) : elle ne doit pas se limiter au seul document formalisé car celui-ci est pratiquement incompréhensible si on le présente seul. Un expert en UML sera capable dans certains cas de reconstituer les intentions initiales en lisant le modèle, mais pas toujours ; et les experts en UML sont rares. Voici la liste des documents qui paraissent nécessaires :

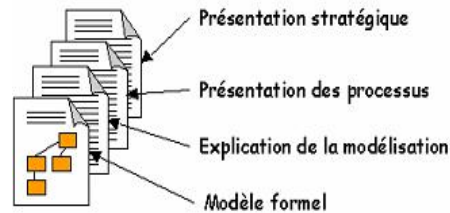


FIG. 9.9 – *Présentation du modèle*

1) présentation stratégique : elle décrit pourquoi l'entreprise a voulu se doter de l'outil considéré, les buts qu'elle cherche à atteindre, le calendrier de réalisation prévu etc. ;

2) présentation des processus de travail par lesquels la stratégie entend se réaliser : pour permettre au lecteur de voir comment l'application va fonctionner en pratique, elle doit être illustrée par une esquisse des écrans qui seront affichés devant les utilisateurs de terrain ;

3) explication des choix qui ont guidé la modélisation formelle : il s'agit de synthétiser, sous les yeux du lecteur, les discussions qui ont présidé à ces choix ;

4) modèle formel : c'est le document le plus épais et le plus difficile à lire. Il est préférable de le présenter sur l'Intranet de l'entreprise : les diagrammes peuvent être alors équipés de liens hypertextes permettant l'ouverture de diagrammes plus détaillés ou de commentaires.

On doit présenter en premier le diagramme d'activité qui montre l'enchaînement des cas d'utilisation au sein du processus, enchaînement immédiatement compréhensible ; puis le diagramme de cas d'utilisation, qui montre le contenu de chaque activité ; puis le diagramme de séquence, qui montre l'enchaînement chronologique des opérations à l'intérieur de chaque cas d'utilisation. Enfin, le diagramme de classes, qui est le plus précis conceptuellement mais aussi le plus difficile à lire : il montre les relations entre classes (agrégation, héritage, association etc.).

* *

Le modèle UML, au moins dans la première étape de son élaboration (modèle métier), transcrit la stratégie de l'entreprise en vue de l'action. Il importe que les abstractions qu'il comporte soient celles qui conviennent au métier et aussi que le métier s'approprie le modèle. La validation du modèle

par le dirigeant du métier (maître d'ouvrage stratégique, voir page 407) est une étape importante de la modélisation : elle permet d'éviter la versatilité des spécifications qui est la plaie des projets.

Il faut pour cela pouvoir présenter au dirigeant le modèle UML sous une forme qu'il puisse lire, et comprendre, ce à quoi la présentation formelle ne se prête pas à l'exception du diagramme d'activité.

L'appropriation collective du modèle par l'entreprise passe par une présentation visuelle du système d'information : on peut ici recommander l'outil OnMap de la société Nomia (www.nomia.com), qui permet de présenter le processus de production comme un dessin animé, selon une mise en scène qui fait comprendre à chacun (au comité de direction tout comme aux agents opérationnels) tout à la fois et le processus, et la façon dont le système d'information l'outil (figure 9.10).



FIG. 9.10 – Un écran d'OnMap

Chapitre 10

La conquête de l'ubiquité

L'ubiquité n'est pas arrivée naturellement à l'ordinateur comme la croissance, puis la maturité, arrivent à un être vivant. Pour les informaticiens des années 1960, l'idée de faire communiquer des ordinateurs n'avait rien d'évident : certains des pionniers de l'informatique lui étaient hostiles car leur intuition leur présentait l'utilisateur et l'ordinateur enfermés ensemble dans une « bulle » douillette ; pour les opérateurs télécoms, l'idée que le réseau puisse servir à autre chose qu'à la téléphonie n'avait rien d'évident non plus. IBM, comme AT&T, furent d'abord hostiles à la mise en réseau des ordinateurs.

La conquête de l'ubiquité résulte des besoins des utilisateurs comme de l'intuition de quelques pionniers et son histoire a comporté une part de hasard. Les architectures, protocoles et conventions qui en résultèrent, et le vocabulaire lui-même, portent la trace de ces hasards tout autant que celle des nécessités.

L'histoire de la conquête de l'ubiquité (Hafner et Lyon [123]) éclaire les unes comme les autres ; la clarté ainsi conquise se projette, au moins qualitativement, sur l'anticipation des évolutions à venir.

* *

10.1 Naissance du réseau

L'ARPA (Advanced Projects Research Agency) a été créée au Pentagone en 1958¹ pour réagir à l'avance scientifique et technique prise par les Soviétiques : le Spoutnik avait été lancé le 4 octobre 1957.

D'abord consacrée à la recherche spatiale, elle fut contrainte dès la création de la NASA à l'été 1958 à se réorienter vers la recherche fondamentale et, pour cela, à s'appuyer sur des partenariats avec les universités.

1. L'ARPA deviendra la DARPA (Defense Advanced Projects Research Agency) en mars 1972. Elle redeviendra l'ARPA en 1993, puis se nommera de nouveau la DARPA à partir de 1996.

En 1962, l'ARPA embauche Joseph Licklider (1915-1990) pour créer en son sein l'IPTO (« Information Processing Techniques Office ») qui animera un programme de recherche en informatique. Licklider était un psychologue devenu expert en informatique. Il lança malgré l'opposition de la profession une recherche sur le temps partagé : « La plupart des constructeurs d'ordinateurs et des directeurs de centres informatiques disaient que le temps partagé entraînait une utilisation inefficace des ressources de la machine, et qu'il ne fallait donc pas l'utiliser »². Ses idées sur la synergie entre l'être humain et l'ordinateur³ conduiront à l'Internet : « Les communautés interactives en ligne se construiront non dans une localisation commune, mais autour de centres d'intérêt communs⁴ ».

En 1964, Licklider fut remplacé par Ivan Sutherland (1938-), lui-même remplacé par Robert Taylor en 1966. Taylor (1932-) restera à la tête de l'IPTO jusqu'en 1969 et jouera un rôle essentiel dans le lancement du programme de recherche sur les réseaux d'ordinateurs⁵, dont il confiera l'animation à Larry Roberts.

Le bureau de Taylor à l'IPTO était connecté *via* trois terminaux différents à des ordinateurs situés à Boston, Berkeley et Santa Monica. Chaque terminal avait sa propre procédure de log-in et ses propres commandes : l'utilité d'une rationalisation sautait aux yeux.

Par ailleurs, en l'absence d'un réseau, chaque centre de recherche qui contractait avec l'ARPA devait acheter et exploiter son propre ordinateur. Il semblait souhaitable de partager les ressources d'une même machine entre plusieurs centres. Mais comment faire, alors que chaque ordinateur utilisait un système d'exploitation et des langages adaptés à ses caractéristiques physiques propres, et ne pouvait communiquer - et encore en mode maître-esclave ! - qu'avec ses propres équipements périphériques ?

Les applications communicantes, comme la messagerie inventée en 1964, restaient limitées aux personnes qui utilisaient un même ordinateur. Il fallait

2. « Most computer manufacturers and directors of computer centres argued that time-sharing was an inefficient use of machine resources and should not be pursued. » (Robert W. Taylor, préface de *In Memoriam J. C. Licklider*, Digital Systems Research Center, 7 août 1990).

3. « Les êtres humains définiront les buts, formuleront les hypothèses, choisiront les critères et réaliseront les évaluations. Les ordinateurs accompliront les tâches routinières nécessaires pour préparer les intuitions et les décisions dans les domaines technique et scientifique » ; « nous espérons que dans peu d'années les cerveaux humains et les ordinateurs seront étroitement couplés, que le partenariat qui en résultera pensera comme aucun cerveau humain n'a jamais pensé, et traitera les données bien autrement que ne le font les machines que nous connaissons aujourd'hui » ; « mieux vaut, pour éviter la polémique avec les partisans de l'intelligence artificielle, leur concéder que dans un futur éloigné la machine aura le monopole de l'intellect. Il n'en restera pas moins un délai confortable pendant lequel les principaux progrès seront réalisés grâce à la collaboration intime entre l'ordinateur et l'être humain » ; « à un ordinateur, il faut indiquer des procédures ; à un être humain, il faut indiquer des buts : les êtres humains semblent penser plus naturellement et plus facilement en termes de buts qu'en termes de procédures. » (Licklider [115]).

4. « On-line interactive communities [...] will be communities not of common location, but of common interest » (Licklider [116]).

5. Cette valse des responsables ne doit pas faire illusion : si les personnes passaient peu de temps à l'IPTO, elles restaient présentes et influentes dans le domaine de recherche.

adapter les ordinateurs à la communication entre égaux, avec les interruptions asynchrones qu'elle comporte ; avant cela, il fallait savoir comment transférer des données à travers un réseau.

10.1.1 La commutation de paquets

Le réseau télécoms était adapté à la conversation téléphonique mais non à la communication entre ordinateurs. Il offrait aux interlocuteurs un circuit bidirectionnel analogique à 4 kHz établi en début de communication et maintenu pendant la durée de celle-ci. Les ordinateurs, qui échangent non des conversations mais des bouffées de données, n'avaient pas besoin d'un circuit permanent ; par contre ils avaient besoin que la transmission des données fût protégée contre les perturbations provenant de l'environnement hertzien et contre les micro-coupures, qui sont peu sensibles en téléphonie mais dévastatrices quand il s'agit de transmettre un flux de données.

La commutation de paquets a été inventée séparément par Paul Baran (1926-), un Américain, et Donald Davies (1924-2000), un Britannique. Avec ce système le réseau transmet non des sons modulant une onde porteuse, mais des bits : il n'est donc plus analogique, mais numérique, ce qui permet d'utiliser des répéteurs informatiques pour corriger les défauts de transmission. Le message est découpé en paquets de taille uniforme contenant l'information nécessaire à leur routage. Enfin, en raison du caractère discontinu des flux de données, une même ressource de transmission peut être utilisée pour plusieurs communications simultanées.

Baran avait étudié la vulnérabilité du réseau à une attaque nucléaire, question cruciale pour les militaires dans la période de guerre froide des années 1960. Il avait découvert que l'on pouvait obtenir une robustesse élevée avec un réseau maillé comportant une redondance relativement faible (il faut relier chaque nœud au réseau par trois ou quatre liens au lieu d'un seul, et équiper chaque nœud d'une table de routage adaptative). Les travaux de Baran sont à l'origine de la légende selon laquelle l'Internet aurait été conçu pour répondre à des besoins militaires : parmi tous les chercheurs qui ont contribué à la mise au point de l'Internet, il semble cependant être le seul qui ait eu cette préoccupation.

Mais AT&T était hostile aux idées de Baran. Au début des années 1960 les commutateurs du réseau téléphonique étaient électromécaniques, la commutation électronique ne devant intervenir que dans les années 1970 ; le circuit établi entre deux interlocuteurs avait ainsi une continuité et une réalité physiques en quelque sorte palpables. Les ordinateurs relevaient d'un autre univers technique que celui des télécoms. « Ils se comportaient comme s'ils savaient tout, et que ceux qui n'appartenaient pas au Bell System ne savaient rien, dira Baran⁶ ; personne d'extérieur à leur système ne pouvait

6. « Their attitude was that they knew everything and nobody outside the Bell System knew anything. And somebody from the outside couldn't possibly understand or appreciate the complexity of the system. So here some idiot comes along and talks about something being very simple, who obviously does not understand how the system works. » ([123] p. 62).

en comprendre la complexité. Et voilà qu'un imbécile s'amène. Il ne comprend visiblement pas comment le système fonctionne, et il prétend que c'est simple! »

Cette phrase illustre les obstacles que rencontrera la mise en réseau des ordinateurs. Les grands du secteur, qu'il s'agisse d'AT&T ou d'IBM, s'opposeront à une innovation qu'ils ne croient pas réalisable ou dans laquelle ils voient une menace : la commutation de paquets rencontrera la même hostilité que le temps partagé.

10.1.2 Le premier réseau d'ordinateurs

Mais comment faire communiquer des ordinateurs (« hosts ») différents, alors que chacun a ses propres caractéristiques physiques, son propre système d'exploitation etc.? Une expérience avait été réalisée en 1965 par le psychologue Tom Marrill qui, sous l'influence de Licklider, avait proposé à l'ARPA de faire communiquer *via* une liaison full-duplex à quatre fils deux ordinateurs de type différent situés l'un à Lincoln, l'autre à Santa Monica.

À cette occasion Marrill mit au point une procédure qui permettait de grouper les caractères dans un message, de les envoyer sur la ligne, de vérifier si le message était arrivé, enfin de le retransmettre s'il n'y avait pas eu d'accusé de réception. Pour désigner cette procédure il retint le mot « protocole » qui aura un bel avenir.

Mais comment faire communiquer non pas deux ordinateurs, mais un réseau de plusieurs ordinateurs? S'il fallait définir un protocole pour chaque couple d'ordinateurs, on était confronté à une complexité quadratique.

Wes Clark proposa de concevoir un petit ordinateur spécialisé, l'IMP (« Interface Message Processor »), qui seul serait mis en réseau et serait interfacé avec chaque ordinateur : cette solution ingénieuse ramenait le problème à la complexité linéaire. Le réseau des IMP fut baptisé « sous-réseau » (subnetwork) pour le distinguer du réseau des ordinateurs eux-mêmes (figure 10.1).

Restaient à régler les problèmes propres à ce sous-réseau : éviter qu'un même paquet ne soit renvoyé sans fin d'un IMP à l'autre, que des paquets ne soient perdus en raison du débordement d'une mémoire (buffer) etc.

Larry Roberts, ayant découvert les travaux de Davies et Baran à une réunion de l'ACM à la fin de 1967, introduisit la commutation de paquets dans l'appel d'offres qu'il lança pour l'IMP en juillet 1968.

Le sous-réseau devait transférer les bits de façon fiable d'un IMP à l'autre; le temps de transit moyen à travers le sous-réseau ne devait pas dépasser la seconde et le sous-réseau devait pouvoir fonctionner de façon autonome, indépendamment des ordinateurs qu'il reliait.

Roberts consulta 140 entreprises. IBM et Control Data répondirent qu'il était impossible de construire un tel réseau car il coûterait trop cher. Finalement BBN (Bolt Beranek and Newman), petite entreprise de Cambridge, sera retenue le 8 septembre 1968. BBN reçut la commande au début de 1969. Sa solution s'appuyait sur le mini-ordinateur DDP-516 d'Honeywell (450 kg, la taille d'un réfrigérateur).

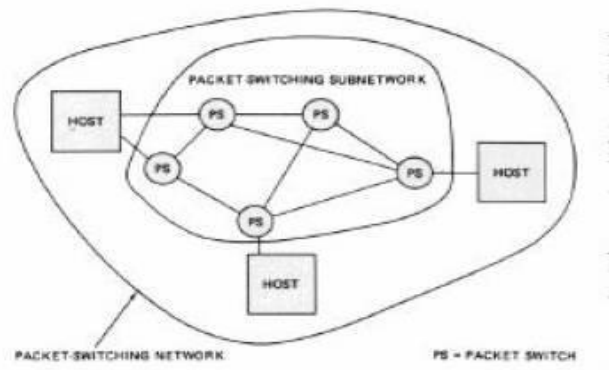


FIG. 10.1 – Réseau et sous-réseau ; source : Cerf et Kahn, [96]

Chaque site équipé d'un IMP devrait produire lui-même, selon les spécifications fournies par BBN, l'interface entre l'IMP et son ou ses ordinateurs (on pouvait raccorder jusqu'à quatre ordinateurs à un même IMP). La mission des IMP étant seulement d'assurer le transport fiable des bits, les ordinateurs devant régler eux-mêmes leurs problèmes de log-in, transfert de fichiers et traitement de texte.

Pour configurer l'IMP BBN préféra le logiciel : une solution matérielle, concrétisée par le câblage, aurait rendu plus rapide l'exécution des tâches simples, mais il aurait été plus difficile de la modifier par la suite.

Le premier IMP est installé à l'UCLA le 30 août 1969 pour raccorder un Sigma 7 ; le deuxième est installé au SRI le 1er octobre pour raccorder un SDS 940. Le troisième est installé à l'UCSB le 1er novembre, le quatrième à l'université d'Utah en décembre. Un « Network Measurement Center » est mis en place à l'UCLA, sous la direction de Leonard Kleinrock, spécialiste de la modélisation et de la simulation des réseaux qui avait ainsi l'occasion de tester ses théories en vraie grandeur.

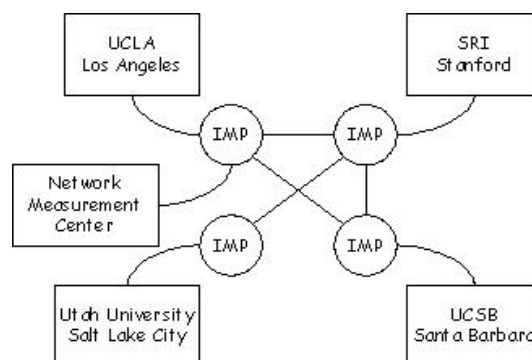


FIG. 10.2 – Le premier réseau d'ordinateurs à la fin de 1969

On remarque sur la figure 10.2 que l'université de l'Utah n'est accessible qu'en passant par le SRI, dont l'IMP joue le rôle d'un routeur.

* *

Les budgets de l'ARPA furent comprimés en 1970 en raison de la guerre du Vietnam, mais les crédits accordés à la recherche en informatique ne furent pas rationnés. En mars 1970, le premier circuit trans-continental est installé vers BBN qui sera le cinquième nœud du réseau. Cela permettra à BBN d'assurer depuis Cambridge la supervision du réseau (télémaintenance, télédistribution des mises à jour par réplication et dissémination). Des indicateurs de qualité étant produits automatiquement par les IMP, BBN pourra détecter les pannes du réseau d'AT&T avant les agents d'AT&T eux-mêmes, ce que ceux-ci auront du mal à accepter.

À l'été 70, le réseau s'étend au MIT, à RAND, SDC et Harvard. Par la suite il s'enrichit d'un nœud par mois. BBN utilisera à partir de 1971 pour les IMP un Honeywell 316, machine plus légère que le 516 et à partir de laquelle est mis au point un TIP (« Terminal IMP ») qui permet de connecter des terminaux au réseau (un TIP peut servir jusqu'à 63 terminaux).

En 1971, le réseau relie 19 ordinateurs et 3 TIP. Pour le désigner, l'expression Arpanet, ou « Net » tout court, apparaît en août 1972.

10.1.3 Problèmes de mise en place

Comme toute nouveauté technique, la mise en place des réseaux d'ordinateurs a rencontré des difficultés imprévues. Le premier IMP livré par Honeywell ne fonctionnait pas et les ingénieurs de BBN ont dû le recâbler à la main. Par la suite, Honeywell s'est de façon persistante refusé à obéir aux spécifications. Le matériel souffrait de pannes aléatoires, d'interruptions asynchrones difficiles à corriger. Pour l'ARPA, 3 % de temps de panne était inacceptable, alors que 97 % du temps en fonctionnement normal semblait à Honeywell être déjà une performance élevée.

Hafner et Lyon notent par ailleurs l'incapacité des « gros » à admettre l'apport de l'innovation, à sortir de l'ornière de leurs habitudes. IBM, nous l'avons vu, a d'abord refusé l'idée de faire communiquer des ordinateurs et AT&T a refusé d'utiliser le réseau télécoms pour transporter des données. En 1971, AT&T refusera encore de prendre la responsabilité de l'exploitation du réseau. En 1983, IBM, DEC et HP préféreront le modèle en couches de l'OSI au protocole TCP/IP. BBN elle-même, entreprise pourtant récente, fut incapable de capitaliser l'avancée acquise avec l'IMP : le directeur du marketing refusa de la lancer sur le marché des routeurs qui fera la fortune de Cisco.

L'ineptie des « gros » est relevée avec délectation par Hafner et Lyon, comme l'avaient fait Carroll [33] à propos d'IBM ou Hiltzik [82] à propos de Xerox. Mais n'est-il pas naturel qu'une grosse entreprise n'ait pas la même capacité de manœuvre qu'une structure légère, et que la procédure de préparation des décisions y soit plus lourde? Le fait est que si les « gros » refusent d'abord l'innovation, ils finissent par « s'y mettre » (avec retard, certes) :

alors seulement les conséquences de l'innovation peuvent se déployer. Ainsi le micro-ordinateur, né en 1973, n'a vraiment percé qu'après le lancement du PC par IBM en 1981.

10.1.4 Protocoles et paradigmes

Le fonctionnement des réseaux nous semble naturel : nous ne percevons pas l'effort intellectuel qui fut nécessaire pour mettre au point les protocoles de communication. À chaque étape, cet effort a été le fait de quelques pionniers qui ont dû lutter pour faire passer une innovation déconcertante ; une fois adoptée, celle-ci s'est imposée comme une norme. Puis elle a résisté aux innovations suivantes, faites par d'autres pionniers.

Le réseau téléphonique était dans les années 1960 entièrement électromécanique, et non pas électronique. Il n'utilisait pas l'ordinateur. Le circuit sur lequel les ondes électromagnétiques porteuses du signal vocal sont acheminées fournissait une continuité physique entre les deux téléphones *via* les lignes d'abonné de la boucle locale, les contacts dans les commutateurs de rattachement et de transit et une bande de fréquences sur le multiplex du réseau de transport.

Le rôle des commutateurs est alors d'établir la communication lors de l'appel, de la maintenir pendant la conversation, puis de libérer les circuits lorsque celle-ci est terminée. Les règles de qualité, d'ingénierie et de dimensionnement de ce réseau étaient définies en fonction du signal vocal et de la matrice de trafic de la téléphonie. Elles délimitaient un univers technique spécifique, d'une grande complexité, dont le fonctionnement supposait la formation et la coopération de plusieurs spécialités.

* *

La commutation de paquets relève d'un univers technique complètement différent. Le transport des données sur les lignes téléphoniques demande une modulation spécifique qui sera faite par les modems⁷ et non plus par les téléphones. Il faut que la qualité des lignes soit suffisante pour transporter des données. Les commutateurs doivent « pédaler » pour lire l'adresse sur chaque paquet, consulter la table de routage, orienter le paquet vers le circuit de sortie convenable. Le circuit transporte d'un commutateur à l'autre non plus une seule conversation, mais des paquets ayant éventuellement des destinations différentes. La statistique du trafic n'est plus la même et il faut des files d'attente (*buffers*) pour stocker les paquets en attente de retransmission.

Les commutateurs électromécaniques étaient incapables de réaliser ces fonctions : il fallait les remplacer par des ordinateurs spécialisés. Les règles de qualité, d'ingénierie et de dimensionnement devaient donc être redéfinies et de nouvelles spécialités professionnelles devenaient nécessaires à l'exploitation du réseau.

7. Les premiers modems avaient été mis au point à la fin des années 1950 pour le système de défense aérienne des États-Unis. Le premier modem commercialisé apparut en 1962 : c'est le Bell 103 d'AT&T qui permet de transmettre 300 bit/s.

En outre, deux techniques entrèrent en concurrence dans l'univers de la commutation de paquets : le « circuit virtuel » et le « datagramme ».

Quand la communication emprunte un circuit virtuel, le premier paquet laisse une trace dans la mémoire des commutateurs qu'il traverse et réserve une capacité de transmission de telle sorte que les paquets suivants puissent emprunter le même itinéraire. Les paquets arriveront ainsi à l'ordinateur destinataire dans l'ordre où ils ont été émis.

Par contre quand on envoie des datagrammes chaque paquet parcourt un itinéraire qui lui est propre, indépendamment des autres paquets ; les délais de transmission étant différents, il se peut que les paquets n'arrivent pas dans l'ordre : le protocole devra permettre de les reclasser à l'arrivée.

Les opérateurs télécoms favorisèrent le circuit virtuel car la continuité qu'il établit à travers le réseau correspond à leur culture professionnelle. Le datagramme, qui suppose entre le réseau et les ordinateurs un partage du travail plus favorable à ces derniers, était par contre bien vu par les informaticiens⁸.

Le protocole X25 a utilisé le circuit virtuel alors que TCP/IP utilisait le datagramme. Pendant longtemps les opérateurs télécoms se méfieront de TCP/IP, qu'ils jugeaient tout à la fois stratégiquement inopportun et peu fiable : cela explique en partie leurs réticences devant l'Internet.

* *

Si les univers de la commutation de paquets et de la téléphonie sont distincts, ils ont en commun la commutation. Celle-ci disparaît dans l'univers des RLPC (réseaux locaux de PC ou LAN, « Local Area Network »), encore plus déroutant pour les gens des télécoms.

Sur un réseau Ethernet, en effet, il n'y a pas de commutateur⁹. Chaque ordinateur est connecté à un « bus » qui lui transmet toutes les trames émises par les autres ordinateurs. Il lit l'étiquette et trie, pour en lire le contenu, celles qui lui sont destinées.

Alors que le réseau commuté met en relation les ordinateurs deux à deux en leur réservant un canal de transmission, le réseau local est donc comme une pièce dans laquelle s'entrecroiseraient plusieurs conversations. Le protocole précise les règles de prise de parole et d'interruption en cas de collision. Il en a existé plusieurs versions (Ethernet, Token Ring etc.) : à chacune correspondent une statistique de trafic et une performance spécifiques (voir page 354).

Pour raccorder deux réseaux locaux, on installe entre eux un pont (bridge) qui trie les trames destinées à l'autre réseau pour les lui faire passer. Si l'on raccorde plusieurs réseaux, il faut un routeur capable d'orienter la trame vers le réseau destinataire. On retrouve donc dans les routeurs une fonc-

8. Les protocoles pouvaient comporter d'autres fonctions : découpage des paquets en trames pour la transmission, et recombinaison des paquets dans chaque commutateur avant réexpédition ; vérification de l'intégrité du paquet à chaque étape, avec ré-émission éventuelle ; etc.

9. Les réseaux Ethernet à haut débit récents peuvent cependant comporter des commutateurs (*switches*) qui accroissent leur efficacité.

tion de commutation, mais elle s'opère entre des réseaux et non entre des terminaux.

Ainsi peuvent se définir des architectures à plusieurs niveaux, la communication entre plusieurs réseaux locaux étant réalisée par une dorsale (*backbone*) à haut débit.

* *

À chacun des types de réseau correspond une statistique de trafic particulière et une définition spécifique de l'encombrement. Sur le réseau téléphonique, l'encombrement se traduit par l'impossibilité d'établir la communication : l'utilisateur reçoit un signal indiquant que les circuits sont occupés et qu'il doit rappeler plus tard. Par contre, sauf accident, une communication en cours n'est jamais interrompue.

Avec la commutation de paquets, l'encombrement se traduit par un débordement des files d'attente dans les routeurs. Il faudra que le routeur, ou l'ordinateur destinataire, envoie un message à l'émetteur pour lui demander d'expédier de nouveau le paquet perdu.

Sur le réseau local, l'encombrement se traduit par de nombreuses collisions entre trames ; si la fréquence des collisions dépasse un certain seuil, le réseau ne peut plus rien transmettre : il est saturé.

Pour chaque type de réseau le dimensionnement doit, lors de la phase de construction, rechercher le compromis raisonnable entre coût et risque d'encombrement. Chaque type de protocole comporte une réponse à l'encombrement : traitement des « tickets d'échec » et filtrage de certains appels dans le cas du réseau téléphonique ; délai de réémission dans le cas de la commutation de paquets et du réseau local. Lors de la conception d'un nouveau protocole, des études statistiques et des simulations sont nécessaires pour vérifier s'il est utilisable et définir ses paramètres. Enfin pour procéder aux ultimes réglages il faut l'expérimenter sur un réseau pilote, puis en vraie grandeur.

Tant que ces études et réglages n'ont pas été faits et que le coût des composants n'a pas été évalué, rien ne garantit que le protocole puisse fonctionner dans des conditions économiques acceptables. C'est pourquoi tout protocole nouveau rencontre, de la part des exploitants, un scepticisme qui ne pourra céder que devant la démonstration et surtout devant l'expérimentation. Aucun protocole ne naîtra s'il n'est pas soutenu par une équipe de pionniers qui, par l'intuition autant que par le raisonnement, anticipent ses performances.

* *

La même communication enchaînera souvent divers protocoles. C'est le cas par exemple d'un ordinateur raccordé à un réseau local et qui consulte un serveur Web : il faut enchaîner Ethernet et TCP/IP. Cela nécessite une passerelle (*gateway*) capable non seulement de lire l'étiquette de la trame pour la faire sortir du réseau local, mais aussi de reconstruire le message pour l'émettre vers l'Internet selon le protocole TCP/IP et inversement dans l'autre sens.

Sur le WAN de l'entreprise (*Wide Area Network*), qui relie les établissements entre eux et aux serveurs informatiques centraux, une cascade de protocoles s'enchaîne à travers des passerelles qui « pédalent » activement lors de chaque communication : Ethernet sur le réseaux local ; IP sur le réseau de raccordement ; X25 ou relais de trame (*Frame Relay*)¹⁰ sur la dorsale intermédiaire ; ATM¹¹ sur la boucle de la dorsale centrale.

10.1.5 TCP/IP et Ethernet

Le mot « protocole » avait été créé par Tom Marrill en 1965 (voir page 333). La conception des réseaux d'ordinateurs demandera la mise au point de plusieurs protocoles. BBN mit au point un protocole de supervision (« remote control ») comportant des outils de diagnostic et de débogage à distance, ainsi qu'un protocole de routage dynamique qui permettra aux paquets de contourner les nœuds et les liens saturés.

Dès l'été de 1968, un groupe d'étudiants des quatre universités qu'il était prévu d'équiper commença à se réunir pour préparer la mise en réseau des ordinateurs. Steve Crocker, de l'UCLA, fut volontaire pour rédiger les comptes rendus. Il publia le 7 avril 1969 une « Request for Comments » (RFC). Cette formule modeste sera conservée par la suite : lorsque les utilisateurs du réseau s'organiseront en un « Network Working Group » (NWG), les RFC resteront le support privilégié de la documentation et de la mise en forme des protocoles¹².

Le NWG décida de définir des protocoles qui traiteraient chacun un problème limité et seraient articulés entre eux : c'était le début du « modèle en couches », innovation majeure en modélisation (voir page 42). Le NWG définit ainsi à la fin de 1969 le protocole Telnet, qui définissait les mécanismes de base de la communication entre deux ordinateurs (connexion et choix des caractères). À l'été 1971, après beaucoup de discussions et de RFC, sortit le « Network Control Protocol » (NCP), protocole de communication sur un réseau d'ordinateurs.

Mais le NCP, qui ne concernait que l'Arpanet, ne permettait pas de faire communiquer entre eux des réseaux différents. Or des projets de réseaux émergeaient en Grande-Bretagne et en France, animés respectivement par Donald Davies et Louis Pouzin. Par ailleurs les réseaux qui utilisaient le satellite, la radio ou, comme l'Arpanet, des lignes téléphoniques, devaient obéir chacun à des contraintes différentes. Il en résultait des choix différents pour la taille maximale des paquets, la vitesse de transmission et les niveaux de fiabilité. Comment construire un « réseau de réseaux » ?

10. Frame Relay (voir www.guill.net/index.php?cat=5&arc=3 est une version allégée de X25 : en simplifiant le protocole et en supprimant des contrôles redondants, elle fait gagner un ordre de grandeur en vitesse de commutation.

11. Le protocole ATM (« Asynchronous Transfer Mode », 1989), conçu par Jean-Pierre Coudreuse pour les réseaux numériques multiservices, est utilisé aujourd'hui dans les dorsales à haut débit.

12. La première RFC est « Host Software », par Steve Crocker, UCLA, 7 avril 1969. La collection complète des RFC se trouve à l'adresse www.faqs.org/rfcs/.

Il fallait définir le protocole qui permettrait de faire communiquer des ordinateurs connectés à divers réseaux. Vint Cerf et Bob Kahn publièrent en mai 1974 un article (Cerf et Kahn [96]) où ils proposaient de découper les messages en datagrammes transmis indépendamment chacun selon son propre itinéraire, la responsabilité de leur reclassement dans le bon ordre incombant à l'ordinateur destinataire. Dans le même article ils introduisirent la notion de passerelle (« gateway ») : une passerelle ne lirait que l'enveloppe du datagramme, seul l'ordinateur destinataire lirait son contenu ; de plus, la passerelle apparaîtrait à chaque réseau comme un ordinateur connecté à ce réseau, et assurerait si nécessaire la transformation du format du datagramme.

Cerf et Kahn appelèrent ce protocole « Transmission Control Protocol », ou TCP. La figure 10.3 transcrit un de leurs schémas.

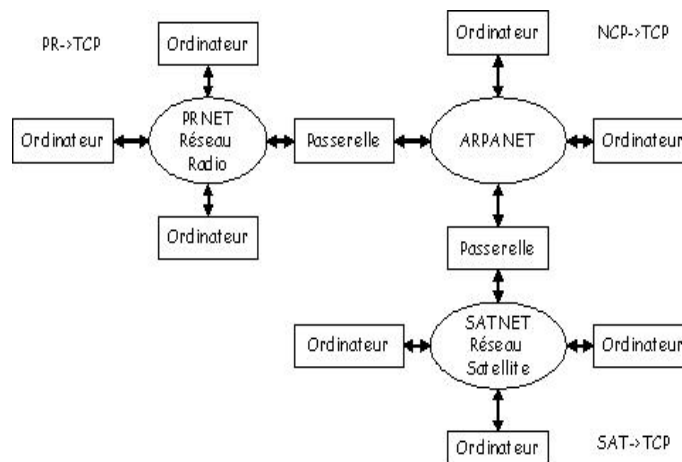


FIG. 10.3 – Le réseau selon le protocole TCP

Au début de 1978, il parut nécessaire de séparer de TCP la partie consacrée au routage des datagrammes, que l'on nommerait IP (« Internet Protocol »), TCP ne traitant plus désormais que ce que font les ordinateurs émetteur et récepteur (découpage du message en datagrammes, mise en ordre des datagrammes à la réception et reconstitution du message, détection des erreurs, réexpédition des datagrammes perdus).

La séparation des deux protocoles permit de construire des passerelles rapides et relativement peu coûteuses, consacrées exclusivement au routage des datagrammes selon le protocole IP. En 1978, TCP devint officiellement TCP/IP. Arpanet abandonnera NCP pour adopter TCP/IP le 1er janvier 1983.

Mais en 1988 l'ISO (« International Standard Organization ») publia le modèle OSI (« Open Systems Interconnection »), longuement attendu par les constructeurs informatiques et les opérateurs télécoms et qui avait leur préférence. Une bataille s'engagea. Les partisans du modèle OSI considéraient TCP/IP comme un bricolage d'universitaires peu conscients des

contraintes de l'économie et de l'industrie. Les partisans de TCP/IP considéraient le modèle OSI comme un produit bureaucratique élaboré à coups de compromis par un comité.

Mais alors que le modèle OSI n'existait que sur le papier TCP/IP fonctionnait et, s'étant forgé dans la pratique, avait incorporé une riche expérience. Un facteur décisif pour le succès de TCP/IP fut son adoption par Sun, entreprise créée en 1982 pour commercialiser des stations de travail Unix : les stations Sun étaient équipées d'une version d'Unix qui incluait gratuitement TCP/IP, ce qui réduisait le coût de leur mise en réseau.

Ethernet sera un autre facteur de succès.

* *

Robert Metcalfe avait, lorsqu'il était étudiant à Harvard, préparé une thèse sur la commutation de paquets en s'appuyant sur l'exemple de l'Arpanet. Harvard avait jugé ce travail trop peu théorique. Metcalfe fut néanmoins embauché par le PARC de Xerox. En 1972, il découvrit le papier qu'avait écrit Abramson [2] sur le réseau Alohanet mis en place à Hawaï grâce à un financement de l'ARPA. Le protocole Aloha était fondé sur une idée originale : au lieu d'être routés d'un ordinateur à l'autre, les paquets étaient émis par radio ; chaque ordinateur recevant tous les paquets, il lui incombait de trier ceux qui lui étaient destinés. Cela permettait de faire communiquer des ordinateurs situés sur des îles différentes de l'archipel (voir page 355).

Metcalfe se fit envoyer à Hawaï par Xerox pour étudier le fonctionnement d'Aloha. Il en améliora la modélisation mathématique, fondée sur le calcul des probabilités, et cela lui permit d'introduire dans sa thèse assez de théorie pour qu'elle soit acceptée par Harvard.

Cependant le PARC avait mis au point l'Alto, machine qui préfigurait l'ergonomie des futurs ordinateurs personnels, et souhaitait mettre les Altos en réseau. Ce travail fut confié à Metcalfe. Équiper chaque Alto d'un IMP aurait été d'un coût prohibitif. Metcalfe utilisa une version améliorée d'Aloha : en faisant passer le signal par un câble coaxial et non par l'espace hertzien, il améliorait le débit du réseau ; en introduisant la détection des collisions, il améliorait le rendement du protocole.

Le premier réseau Ethernet fut ainsi mis en place en 1973 au PARC. Ses spécifications ne seront rendues publiques que le 30 septembre 1980 et la norme IEEE 802.3 qui en est issue ne sortira qu'en 1983. Les réseaux locaux se répandront dans les entreprises à partir de 1989 avec le lancement d'Ethernet 10BaseT, qui permet de transporter le signal sur une paire torsadée semblable à celle du réseau téléphonique d'un établissement.

Dès lors les réseaux d'ordinateurs vont pouvoir s'appuyer à la fois sur TCP/IP et sur Ethernet. Ethernet sert à la communication entre les ordinateurs connectés à un même réseau local ; TCP/IP assure la communication à distance. Ethernet est analogue à une conversation dans une salle où chacun prend la parole quand il a quelque chose à dire, s'interrompant en cas de collision avec un autre intervenant ; TCP/IP est (en plus rapide) semblable à l'envoi d'un texte, *via* la poste, par morceaux successifs qu'il faut classer à l'arrivée.

Physiquement, le support de l'Ethernet peut être divers : paire torsadée, câble coaxial, fibre optique, espace hertzien des réseaux WiFi ou encore câblage électrique de l'immeuble. L'Internet, lui, est composé d'un ensemble de routeurs et de liaisons louées aux opérateurs télécoms. Un réseau local Ethernet sera relié à l'Internet de préférence par une passerelle et une liaison louée. Ainsi, et sous réserve des droits d'accès de son utilisateur, n'importe quel ordinateur d'un établissement peut communiquer avec un autre ordinateur d'un autre établissement, la communication empruntant des passerelles entre les réseaux Ethernet et TCP/IP (figure 10.4).

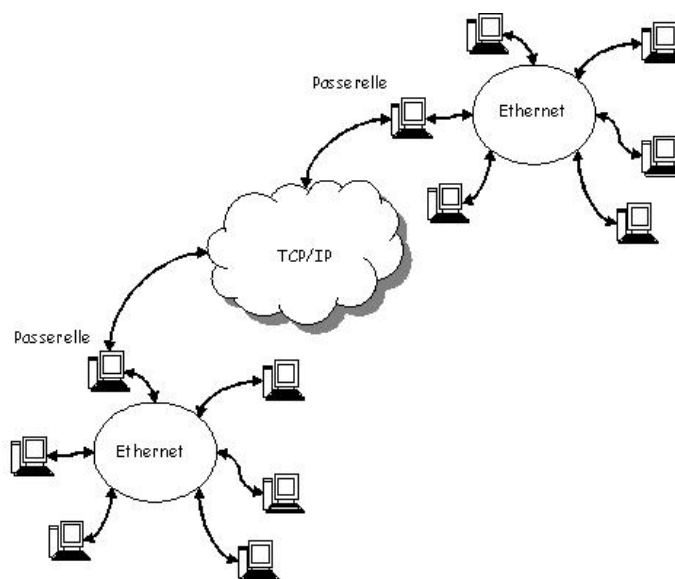


FIG. 10.4 – *Ethernet et TCP/IP*

L'utilisateur individuel à domicile passe, lui, par le modem et la ligne téléphonique ou par l'ADSL pour se relier à un fournisseur d'accès à l'Internet (FAI), lui-même relié au réseau TCP/IP par une liaison louée.

Pour des raisons de sécurité certaines entreprises cloisonnent leur réseau : elles utilisent des réseaux privés virtuels fortement protégés (RPV, ou VPN pour *Virtual Private Network*) selon des architectures conçues par les opérateurs télécoms. Sur ces réseaux peuvent entrer en jeu d'autres protocoles que TCP/IP (X25, Frame Relay, ATM etc.).

Ces entreprises définissent aussi un *Intranet* fournissant à l'intérieur de l'entreprise des services analogues à ceux que l'on trouve sur l'Internet (messagerie, documentation électronique, moteur de recherche etc.). Alors le schéma se diversifie et s'enrichit, mais l'articulation entre le protocole du réseau local et celui (ou ceux) du réseau de transport reste la règle.

10.1.6 Premières applications

Les applications qu'il était prévu de mettre en œuvre sur l'Arpanet étaient le login à distance et le transfert de fichier, mais le cours des événements apportera des surprises.

Le tout premier essai, après l'installation du deuxième IMP en octobre 1969, consista à simuler (mais à distance) la connexion à l'ordinateur d'un terminal « bête », c'est-à-dire dépourvu de mémoire et de processeur. Le login à distance fut formalisé en décembre 1969 par le protocole Telnet qui permettait d'établir la connexion et déterminait le jeu de caractères à utiliser.

Mais pour échanger des fichiers entre ordinateurs, qu'il s'agisse de programmes ou de données, il fallait disposer d'un protocole de transfert de fichiers : seul un tel protocole permettrait à deux machines de coopérer d'égal à égal au lieu que l'une soit comme un terminal de l'autre. La mise au point de ce protocole fut difficile en raison des différences entre machines. FTP (File Transfer Protocol) ne sera disponible qu'en juillet 1972.

À l'automne de 1971, en l'attente de FTP, l'Arpanet ressemblait à une autoroute sans automobiles : le réseau, utilisé à 2 % de sa capacité, n'était qu'un banc d'essai pour des expérimentateurs qui, afin de tester sa robustesse, le poussaient de temps à autre dans ses retranchements en générant un trafic artificiel.

En octobre 1972 enfin, lors de la première ICCC (*International Conference on Computer Communication*), la DARPA put mettre en scène une démonstration *via* un TIP connecté par deux liens à 50 kbit/s. Grâce à Telnet et à TCP, plus de quarante terminaux installés à l'hôtel Hilton de Washington purent ce jour-là se connecter à 29 ordinateurs disséminés sur le territoire américain pour utiliser diverses applications. Les professionnels découvrirent alors que la commutation de paquets fonctionnait et que des ordinateurs pouvaient communiquer pour échanger des données, programmes et résultats. De grandes bases de données pourraient ainsi être mises à disposition à distance. La perspective d'un marché nouveau éveilla l'appétit.

Par ailleurs même si seuls les sites qui avaient un contrat avec la DARPA pouvaient communiquer sur l'Arpanet, la diversité des personnes qui travaillaient dans les universités élargissait le cercle des utilisateurs au delà de la communauté des spécialistes de l'informatique. Une revue mensuelle, *Arpanet News*, leur procura à partir de 1973 la liste ce que chaque site avait à offrir.

L'Arpanet fournissait donc désormais le service pour lequel il avait été conçu, même si son manque d'ergonomie le réservait aux personnes qui s'y connaissaient assez en informatique. Cependant la majorité du trafic fut bientôt absorbée par un service que personne n'avait prévu : la messagerie.

* *

Dès les années 1960 les utilisateurs d'un ordinateur en temps partagé avaient utilisé un service de messagerie comme Mailbox : mais ce service fonctionnait autour d'un même ordinateur, donc dans le même immeuble,

pour des personnes qui n'auraient eu que quelques mètres à parcourir pour se parler.

Le service change de nature lorsque la messagerie est utilisée par des personnes que séparent des centaines ou milliers de kilomètres. Alors que la messagerie sur *mainframe* n'avait été qu'un jouet amusant et commode on découvrit que l'Arpanet, en effaçant la distance géographique, en faisait un outil des plus utiles.

Il fallait d'abord disposer d'un logiciel de messagerie. Le premier fut mis au point par Ray Tomlinson, qui eut en 1972 l'idée d'articuler le programme de messagerie d'un ordinateur en temps partagé avec un protocole de transfert de fichiers de telle sorte que l'on puisse échanger des messages entre divers ordinateurs. Cette possibilité, jugée intéressante par les rédacteurs du protocole FTP, y fut insérée en août 1972.

Mais utiliser l'Arpanet pour transmettre des messages personnels semblait quelque peu illicite : le réseau n'avait pas été fait pour cela. Par ailleurs il n'était pas très facile en 1972 d'envoyer un message sur l'Arpanet : il faudra du temps pour surmonter les difficultés techniques et la messagerie ne sera d'un usage commode que vers 1980. Son caractère licite sera peu à peu reconnu, mais les questions de savoir-vivre qu'elle pose susciteront des discussions passionnées.

Tomlinson est resté célèbre pour une décision qu'il prit en écrivant son programme (Le Diberder [48]). Il fallait, pour séparer dans l'adresse d'un message le nom de l'utilisateur de celui de la machine sur laquelle il travaillait, un caractère qui ne puisse jamais apparaître dans un nom propre. Tomlinson remarqua sur son clavier le symbole @, caractère typographique rare, et décida de le retenir. L'arobase est devenu le symbole de la messagerie électronique mais cette convention gênera les utilisateurs de Multics, système d'exploitation où @ signifie « supprimer la ligne ».

Stephen Lukasik, directeur de la DARPA de 1971 à 1975, fut l'un des plus actifs parmi les promoteurs de la messagerie électronique. Bien vite ses collaborateurs comprirent que la messagerie était le moyen le plus efficace pour communiquer avec lui et obtenir une décision sur leurs projets. Ils rencontraient ainsi un phénomène qui se répétera souvent : pour que la messagerie se répande dans une entreprise, il faut que le patron lui-même l'utilise activement.

Dès 1973 la messagerie représentait les trois quarts du trafic de l'Arpanet. Mais s'il était facile d'envoyer un message il n'était pas commode de l'écrire, de le lire, moins encore de lui répondre. Les outils de traitement de texte étaient rudimentaires et à la réception les messages s'affichaient à la queue-leu-leu sans que rien ne les sépare : il fallait les parcourir tous. Comme il n'existait pas encore d'instruction « Réponse », il fallait pour répondre à un message composer un nouveau message.

Plusieurs programmeurs entreprirent de combler ces lacunes. Les programmes de gestion de messages se multiplièrent. Il en résulta pour les opérateurs une telle complexité que bientôt le besoin d'une normalisation pour le plan d'adressage, le codage des dates etc. devint évident. Mais quand on veut normaliser la messagerie on touche à des valeurs enfouies dans les

consciencés et cela déclenche des conflits. La définition de l'en-tête (*header*) occasionna ainsi une bataille de plus de dix ans entre ceux qui préféraient un en-tête sobre et ceux qui voulaient le truffier d'informations techniques¹³. La plupart de ces conflits furent arbitrés au sein d'un groupe de travail *ad hoc*, le MsgGroup, actif de 1975 à 1985 (les listes de diffusion ont été inventées pour faciliter le fonctionnement du MsgGroup).

John Vittal introduisit en 1975 dans son programme MSG l'instruction « réponse » (*Answer*) ainsi que des outils permettant de gérer le flux des messages reçus. MSG fera beaucoup pour la commodité et la popularité de la messagerie.

On ne savait pas encore que la messagerie était un amplificateur d'agressivité¹⁴. Les engueulades (*flaming*) devinrent fréquentes au sein du MsgGroup, par exemple entre ceux qui travaillaient à la normalisation des entêtes, ou encore entre ceux qui étaient pour ou contre les outils qui permettent à chaque instant de savoir qui est en ligne. Par ailleurs des lecteurs s'offusquaient de messages conçus comme des plaisanteries. Kevin MacKenzie, soucieux de retrouver l'expressivité qui pondère le langage oral, proposa en avril 1979 de compléter la ponctuation par le symbole « :-) ».

En 1981 le protocole de transfert de messages (MTP) (*Message Transfer Protocol*) inclus dans FTP ne suffisait plus. John Postel mit au point SMTP (*Simple Message Transfer Protocol*) qui traite l'envoi des messages et leur transfert du serveur de l'expéditeur au serveur du destinataire. La première version de POP (*Post Office Protocol*), qui permet au destinataire de retirer ses messages sur le serveur après s'être identifié, fut publiée en 1984.

Dans le courant des années 1980, le MsgGroup devint de moins en moins productif et finalement il disparut : il avait fait son travail. La messagerie existait telle que nous la connaissons aujourd'hui, avec toutes ses possibilités et aussi avec les problèmes de savoir-vivre qu'elle pose et qui ne sont pas encore maîtrisés (voir par exemple Harmon [79]).

Cette innovation inquiéta l'US Postal Service. Certes le volume des messages restait très inférieur à celui du courrier sur papier mais son taux de croissance était impressionnant et, dès 1976, la messagerie commençait à être utilisée en dehors du cercle des chercheurs. Une étude d'Arthur D. Little disait qu'elle absorberait en quelques années 30 % du courrier urgent.

En 1979, le Postal Service tenta de la concurrencer en offrant un service hybride inspiré du télégraphe : les messages transiteraient d'un bureau de poste à l'autre pendant la nuit, seraient imprimés à l'arrivée, puis livrés à domicile par porteur le lendemain matin. Ce projet coûteux et bizarre, mais significatif de la rigidité des grosses institutions, fut bientôt abandonné.

13. Certains voulaient introduire dans l'en-tête non seulement le nom de l'expéditeur et la date d'envoi, mais aussi le nombre de caractères, des mots clés, le niveau de sécurité, l'identité de la machine etc. Une partie de ces informations se trouve, aujourd'hui, dans l'en-tête caché de nos messages (pour les lire avec Outlook il faut choisir « Options » dans le menu « Affichage » du message).

14. « The speed of e-mail promoted flaming » (Hafner et Lyon [123] p. 216).

Certains des ingénieurs qui coopéraient à la conception de l'Arpanet s'étaient, dans leur temps libre, passionnés pour les jeux de société « Donjon & Dragon », mondes imaginaires où chacun joue un rôle de son invention. L'un d'entre eux, Will Crowther, travaillait à BBN. C'était par ailleurs un amateur de spéléologie. Pour amuser ses enfants il modélisa en 1976 un réseau de souterrains et y plaça une version simplifiée de Donjon & Dragon qu'il baptisa « Adventure ». Il abandonna bientôt ce brouillon écrit en quelques week-ends mais une copie fut retrouvée par Don Woods sur un ordinateur de Stanford. Woods, après avoir retrouvé Crowther au PARC par une recherche sur la messagerie, obtint le code source et l'autorisation de le modifier. Il perfectionna le jeu et selon la plus pure tradition du logiciel libre le mit à disposition sur l'ordinateur du laboratoire d'intelligence artificielle de Stanford.

Les copies d'Adventure se multiplièrent sur l'Arpanet et de nombreux joueurs s'y consacrèrent avec passion. Découvrir que l'on pouvait non seulement travailler, mais jouer avec un ordinateur suscita de nombreuses vocations de programmeur (Bennahum [10]). L'Arpanet contribua ainsi à la naissance de l'industrie du jeu sur ordinateur, et même du jeu en réseau.

10.1.7 Vers l'Internet

Pour construire l'Arpanet, la solution avait consisté à louer des lignes téléphoniques à AT&T. Mais l'ARPA chercha à diversifier les modes de transmission. En 1969, Bob Taylor avait accordé un budget à l'Alohanet, réseau créé par Norman Abramson à l'université d'Hawaï et qui utilisait la transmission hertzienne. Une version radio de l'Arpanet fut déployée après 1972.

Si l'on voulait utiliser la radio pour transmettre des paquets de données entre des sites éloignés, en particulier les sites mobiles comme les bateaux ou les chars de combat, la meilleure solution serait de passer par les satellites de télécommunications. Il en résulta la mise au point du réseau Satnet.

Mais chaque réseau de transport - lignes téléphoniques, radio, satellite - doit obéir à des contraintes physiques qui lui sont propres. Il en résultait des spécifications différentes de la taille des paquets, des délais de retransmission en cas d'échec etc. Il fallait qu'une même communication pût emprunter ces divers réseaux sans que les ordinateurs qui voulaient communiquer n'aient à se soucier de ces complications.

La diversification des réseaux n'était pas seulement technique mais aussi géographique. En France, Louis Pouzin avait créé Cyclades pour relier les universités. Donald Davies avait lancé un réseau de commutation de paquets en Grande-Bretagne. Un *International Network Working Group* (INWG) fut donc créé en 1972 pour organiser l'interconnexion des divers réseaux à travers le projet *Concatenated Network*, ou Catenet. En 1973, l'ARPA créa à cette fin un *Interneting Project*.

Il fallait remplacer le *Network Control Protocol* (NCP) de l'Arpanet par un protocole permettant de communiquer à travers plusieurs réseaux différents. L'article de Vint Cerf et Bob Kahn de mai 1974 [96] définit le

Transmission Control Protocol (TCP qui deviendra ensuite TCP/IP, voir page 340). Chaque réseau pourrait être exploité indépendamment, selon son propre protocole et par ses propres opérateurs ; des passerelles (*gateways*) prendraient en charge la communication entre réseaux différents et garantiraient la continuité de la communication d'un bout à l'autre. L'astuce était de faire assurer la fiabilité de la communication par les ordinateurs émetteur et destinataire, le rôle du réseau étant seulement de transporter les datagrammes.

* *

À la fin des années 1960, la guerre du Vietnam avait suscité entre l'armée américaine et la société civile des tensions qui se manifestaient vivement dans les universités. On soupçonnait l'armée d'utiliser l'Arpanet pour collecter des informations et fichier les opposants, ou encore pour des recherches sur l'utilisation stratégique de l'arme nucléaire. Par ailleurs l'exploitation du réseau se compliquait en raison de sa croissance. L'armée se désengagera donc progressivement de l'Arpanet, le laissant évoluer vers l'Internet selon un processus qui comportera quelques épisodes paradoxaux.

Il fallait ainsi trouver un sous-traitant pour assurer l'exploitation quotidienne du réseau, mais avant de lâcher la main l'armée voulut s'assurer qu'elle ne perdrait pas tout contrôle sur une structure qui pouvait avoir un intérêt stratégique. La responsabilité du réseau quitta donc la DARPA en 1975 pour passer à la DCA, « Defense Communications Agency », organisme purement militaire qui, de surcroît, partageait le scepticisme d'AT&T envers la commutation de paquets. Une bureaucratie de généraux et de colonels s'installa, prescrivant en détail et par écrit ce qu'il fallait faire et comment le faire.

Il était naturel de passer à BBN le contrat relatif à l'exploitation du réseau mais BBN était entrée en conflit avec la DARPA en refusant de communiquer le code source de l'IMP, ce qui gênait beaucoup ceux qui devaient corriger les dysfonctionnements du réseau. Ce code ayant été financé par le budget fédéral, la revendication de BBN parut excessive. La DARPA menaça de lui couper tous ses contrats. BBN accepta enfin de fournir pour une rémunération symbolique le code source à qui en voudrait ; la DCA lui confia alors l'exploitation opérationnelle du réseau.

En 1975, les spécifications techniques de TCP/IP sont disponibles ; Vint Cerf arrive à la DARPA en 1976 pour prendre la responsabilité de l'interconnexion des réseaux Arpanet, Satnet et radio, projet nommé « ARPA Internet ». En octobre 1977 cette interconnexion fonctionnait.

* *

La NSF (National Science Foundation), créée en 1950 pour promouvoir le progrès scientifique en finançant la recherche fondamentale et la formation, avait dès 1974 compris l'intérêt d'un réseau pour l'accomplissement de sa mission. Mais pour disposer d'un site Arpanet une université devait avoir un contrat avec la DARPA sur des projets de recherche financés par la Défense. Un site Arpanet coûtait plus de 100 000 \$ par an à cause du coût des liaisons louées. En 1979, 120 *Computer Science Departments* étaient en

place dans les universités américaines mais celles-ci n'exploitaient que 15 des 61 sites Arpanet. L'université risquait donc de se scinder en deux, une cloison se créant entre celles qui étaient connectées et celles qui ne l'étaient pas. Larry Landweber, de l'université du Wisconsin, proposa de créer un réseau ouvert à la recherche, l'université et l'industrie en louant des liens à Telenet, filiale de BBN qui exploitait un réseau de commutation de paquets plus lent que l'Arpanet mais moins coûteux.

En 1980, la NSF accepta de financer pour cinq ans un *Computer Science Research Network* (CSNET). En 1986, pratiquement tous les départements d'informatique des universités et beaucoup de centres de recherche privés étaient connectés au CSNET et le coût de son exploitation était équilibré par les redevances de ses utilisateurs. À l'exemple du CSNET, d'autres réseaux se mirent en place dans les années 1980 : BITNET (« Because It's Time Network ») entre les systèmes IBM ; UUCP (« Unix-to-Unix Copy Program ») aux Bell Labs ; SPAN (« Space Plasma Analysis Network ») à la NASA, et divers réseaux universitaires en Europe et au Canada. Ces réseaux communiquaient en utilisant TCP/IP. On commença alors à utiliser le terme « Internet » pour désigner l'interconnexion mondiale des réseaux TCP/IP. L'industrie des routeurs prit son essor.

En 1985, la NSF accepta de construire entre cinq ordinateurs répartis sur le territoire américain la dorsale NSFNET, réseau à haut débit auquel les réseaux régionaux des universités pourraient se connecter (il est économiquement efficace d'organiser un réseau en plusieurs niveaux, le niveau fédérateur fournissant le débit le plus élevé : une dorsale permettait de diminuer le coût du réseau). Sa disponibilité suscita la création de plusieurs réseaux régionaux : NYSERNET à New York, CERFnet en Californie etc. La NSF finançait les premières années d'exploitation d'un réseau universitaire, après quoi chaque université devait payer 20 000 à 50 000 \$ par an pour une connexion à haut débit.

Il fallait identifier les ordinateurs connectés à l'Internet. Le *Domain Name System* (DNS), défini en novembre 1983, proposa une structure d'adressage hiérarchique. Sous la pression de la DARPA ce système fut adopté par l'ensemble des acteurs en janvier 1986, les sept domaines de haut niveau étant edu, com, gov, mil, net, org et int.

En 1989, l'Internet démarre au plan économique¹⁵. TCP/IP s'impose dans le monde entier : l'Internet n'est plus une constellation d'ordinateurs centrée sur l'Arpanet mais un ensemble de réseaux connectés à la dorsale du NSFNET, vingt-cinq fois plus rapide que l'Arpanet et beaucoup plus commode. L'Arpanet n'était plus désormais que l'un des réseaux Internet de la DARPA. Il ne restait qu'à débrancher l'un après l'autre les IMP pour faire basculer chaque ordinateur vers un des réseaux régionaux de l'Internet. À la fin de 1989 ce fut chose faite, non sans quelque nostalgie.

15. « We started looking at the network statistics and realized we had a rocket on our hands », « Nous nous mîmes à regarder les statistiques de trafic, et réalisâmes alors que nous tenions une fusée dans nos mains » (Vint Cerf, cité dans Hafner et Lyon [123] p. 254).

En 1991 naissait la Toile (Tim Berners-Lee [11]). En 1993 apparaissait Mosaic, le premier navigateur : l'Internet était ainsi doté des facilités ergonomiques qui avaient fait défaut à l'Arpanet.

En 1995, l'administration américaine interrompit le financement du NSF-NET : l'Internet devenait une affaire purement commerciale, rémunérée par le paiement des utilisateurs aux fournisseurs d'accès qui eux-mêmes paient leur raccordement, ce qui finance de proche en proche une architecture à base de liaisons louées, routeurs et dorsales.

L'évaluation économique prouve alors que l'Internet est viable grâce à la simplicité de l'architecture que permet le protocole TCP/IP et à la baisse tendancielle du coût des équipements électroniques¹⁶. Cette baisse facilite l'accès à l'Internet d'une population d'utilisateurs qui s'élargira, au delà du cercle des spécialistes de l'informatique, universitaires et chercheurs, pour inclure potentiellement l'ensemble de la population et lui fournir des services multimédia.

De même qu'un cintre sert à construire la voûte qui, une fois posée, tient toute seule, l'Arpanet a servi à construire un être nouveau, techniquement efficace et économiquement viable. Les projets de Licklider, utopiques au début des années 1960, étaient dans les années 1990 devenus une réalité. Les institutions, sceptiques dans un premier temps, se rallieront l'une après l'autre à ce nouveau média - mais elles devront s'y adapter.

10.2 Naissance de la Toile

« Computers help if we use them to create abstract social machines on the Web : processes in which the people do the creative work and the machine does the administration. » (Tim Berners-Lee [11] p. 172).

La Toile consiste en la fusion des apports de l'hypertexte et de l'Internet. Avant la Toile, l'hypertexte existait, mais seulement à l'intérieur d'un même corpus de documents ; l'Internet existait, mais il ne permettait pas la navigation commode dans un ensemble de documents. Pour les faire fusionner il a fallu bâtir un édifice technique comportant trois éléments : l'URL (« Uniform Resource Locator », adresse unique d'un document) ; HTML (« Hypertext Markup Language »), langage simple de mise en forme des documents ; HTTP (« Hypertext Transfer Protocol »), protocole de communication pour l'échange des documents. XML (« eXtensible Markup Language ») succédera ensuite à HTML et enrichira la Toile en permettant d'articuler les données structurées et le langage naturel. Dans le monde du logiciel, il s'est bâti des édifices autrement imposants ! Mais ici il s'agissait de changer la façon dont les ordinateurs sont utilisés : c'est du côté de l'usage que se trouvait la difficulté essentielle.

16. Talière et Volle [210]

Lorsqu'il était enfant, Tim Berners-Lee avait eu une conversation avec son père, mathématicien, qui préparait un article sur les relations entre l'ordinateur et le cerveau. L'enfant en avait retiré une conviction durable : « l'ordinateur pourrait être beaucoup plus puissant si l'on pouvait le programmer de telle sorte qu'il puisse relier des informations que rien n'interconnecte par ailleurs ». Cette conviction est à l'origine de la Toile : il s'agissait de mettre l'informatique au service des associations d'idées, de cette étape obscure où l'intuition prépare à tâtons la construction conceptuelle.

Au CERN, centre de recherche sur la physique des particules, Berners-Lee a trouvé un terrain à la fois favorable et indifférent. Favorable parce que l'écheveau des collaborations et des expériences qui se noue au CERN pose de multiples problèmes d'interopérabilité entre systèmes d'information ; aussi parce que comme dans tout bon centre de recherche il y est possible, dans certaines limites, de consacrer du temps de travail à un projet que personne ne comprend.

Indifférent parce que même si le CERN a utilisé à partir de 1991 les premières versions de la Toile il ne s'est jamais passionné pour un projet qui, il est vrai, ne relevait pas de son activité principale.

Le projet, dans ses premières étapes, aurait pu être écrasé par un gestionnaire soucieux de recentrer les efforts du CERN sur leur cible principale ; Berners-Lee le sentait et il a su adapter son travail de sorte qu'il soit immédiatement utile aux projets du CERN, dût-il pour cela renoncer à une définition plus stricte ou plus logique de ses priorités.

Lorsque la Toile a décollé en 1993 elle a rencontré d'autres projets (comme Mosaic, ancêtre de Netscape) qui voulaient la monopoliser. Berners-Lee a dès 1994 mis en place le W3C (*World Wide Web Consortium*) et défini ses règles de fonctionnement. Il a fait cela par tâtonnement, sous l'empire de la conviction ci-dessus et aussi d'un idéal qui est son autre moteur.

Berners-Lee n'a pas en effet pour priorité de s'enrichir, ce qui surprend beaucoup les Américains. Il s'est donné pour but de faire de la Toile un lieu (purement logique, non situé dans l'espace) équipé de tout ce qui peut faciliter le travail coopératif. Il estime nécessaire d'adapter l'ordinateur aux exigences de la pensée : il faut que la machine démarre instantanément et que la recherche sur la Toile ramène immédiatement l'information voulue de telle sorte que la pensée puisse rebondir selon son propre rythme ; il faut que l'hypertexte permette de suivre les associations d'idées qui sont le terreau de la réflexion.

10.3 Lire et publier sur la Toile

Berners-Lee voudrait que l'on pût disposer sur la Toile non seulement d'un texte, mais de ses diverses versions et des commentaires qui lui ont été apportés. Disposer de la version finale d'un texte, c'est bien. Mais disposer aussi des annotations déposées sur ce texte par divers commentateurs, ou de ses versions provisoires, est-ce vraiment mieux ? Un texte n'est lisible que si l'on consent une perte en information : l'imprimé est plus lisible qu'un manuscrit et il ne porte pas l'information qu'apportait l'écriture manuscrite.

Lorsque nous aurons lu un texte, ses annotations, ses versions successives, que restera-t-il de la journée de travail?

Lorsqu'on parle de travail coopératif il faut d'abord en concevoir la forme la plus simple, qui est le dialogue entre auteur et lecteur. Or depuis que les hommes savent parler - un conteur est un auteur - maîtrisent-ils ce dialogue? Savoir parler, savoir écrire, savoir lire : ces trois savoirs à l'énoncé si simple recouvrent en fait des exigences sans limites. Il faut non seulement savoir parler, mais savoir gérer sa parole ; non seulement savoir lire, mais gérer son activité de lecture.

Le « Web sémantique » auquel aspire Berners-Lee pose donc des problèmes de savoir-vivre autant que de savoir-faire. Il reste à définir une hygiène de la Toile, attentive au rythme de la pensée, aux conditions de la concentration, à la formation des images centrales autour desquelles s'articulent représentations et concepts. Comment faire pour que la Toile nous assiste sans nous abrutir? Au « zapping » qui fait « surfer » d'une page à l'autre sans lire et qui suscite l'écœurement, ne préférons-nous pas la recherche des rares textes qui méritent une lecture attentive, lente et concentrée?

Berners-Lee n'évoque pas ces questions-là ; mais personne n'oserait reprocher à Gutenberg de ne pas avoir parlé des problèmes que soulève la diffusion des livres, le choix des lectures, la gestion d'une mémoire que l'imprimé accable autant qu'il la soutient. Le livre articule divers acteurs : auteur, éditeur, imprimeur, distributeur, libraire, lecteur, critiques littéraires, historiens de la littérature, professeurs, jury des prix, journalistes, spécialistes du droit d'auteur etc. Tout ce monde est relié par des rapports économiques et symboliques.

La Toile n'est pas plus spontanée que le livre ; elle a besoin de qualité éditoriale, de critiques, d'animateurs qui puissent fournir à l'utilisateur des points de repères. Les textes ne sont pas de valeur équivalente, tant en ce qui concerne l'écriture que la pensée. Des institutions (non autoritaires, certes) seront nécessaires pour faciliter son utilisation, tout comme le W3C a été nécessaire pour faciliter son éclosion.

Certains se plaignent de la mauvaise qualité des informations que l'on trouve aujourd'hui sur la Toile. Ils ont à la fois tort et raison. Quand on utilise un moteur de recherche comme www.google.fr, et si l'on tape des mots clés bien choisis, arrivent plusieurs dizaines d'URI¹⁷. On clique sur ces adresses et c'est comme si l'on fouillait une poubelle : on trouve l'équivalent d'une chaussure usée, d'un journal de la veille, d'une chose innommable et visqueuse puis pouf ! une pièce d'or - c'est-à-dire la réponse à la question que l'on se posait - et cette événement fait la différence entre la Toile et une poubelle : vous pourriez fouiller longtemps des poubelles, vous n'y trouverez pas souvent des pièces d'or.

Il est vrai que parfois on ne parvient pas à trouver ce que l'on cherche. Il est impossible à ce jour de trouver sur la Toile le contenu de l'ouvrage *History of Programming Languages*, Addison-Wesley 1981, qui rend compte

17. *Uniform Resource Identifier*.

de la conférence HOPL de l'ACM en 1978 et décrit notamment les origines de Fortran, Cobol et LISP : l'ACM diffuse une abondante documentation, mais pas celle-là : « les contenus des ouvrages ne sont pas mis sur la Toile ».

Ainsi l'on trouve sur la Toile des commentaires, parfois erronés, mais il est difficile d'y trouver certains textes originaux. C'est sans doute une question de droits d'auteur. Il est pourtant tout naturel de mettre du contenu sur un site Web ! Où le mettre sinon ? Sur des étagères ? Dans la tête de l'auteur ? Mes « idées », qui sont d'ailleurs souvent celles que d'autres m'ont suggérées, je ne les considère pas comme ma propriété mais comme un flux dont je suis le modeste transmetteur ; et un flux ne présente d'intérêt que s'il circule.

Il faut d'ailleurs relativiser cette question des droits d'auteur. Que rapporte à son auteur un livre sur lequel il aura sué pendant des mois, un livre de réflexion qui, étant sérieux, n'a pas les caractéristiques favorables à la grosse vente ? Il sera imprimé à un petit millier d'exemplaires dont quelques centaines seront vendus. S'il se vend 30 € et si l'auteur a 10 % de droit d'auteur, il lui reviendra 300 € par centaine de livres vendus et ce sera un grand succès s'il lui rapporte 1 500 € par an pendant deux ans. Comparé au prix de journée d'un consultant, et compte tenu du temps que l'on doit consacrer à l'écriture d'un livre, c'est autant dire rien.

On ne publie pas un livre sérieux, sur un sujet auquel on a longuement réfléchi, pour l'argent que cela rapporte mais pour mettre en circulation des idées que l'on croit utiles et pour lesquelles on estime devoir militer. Or l'édition sur papier est de ce point de vue un vecteur peu efficace. Votre livre sera publié parmi des milliers d'autres ; il ne sera présenté sur les devantures des librairies, dans les rayons, que pendant les semaines qui suivent sa publication. Il n'existera donc sur le marché que durant la courte période où il constitue une nouveauté. Il en est de même des articles : vous publiez dans une revue, fort bien ; dès que le numéro suivant paraîtra, votre article commencera à jaunir sur les étagères.

Qu'il s'agisse d'un livre ou d'un article, le rendement de l'imprimé en termes de communication est très limité. C'est pourquoi il est difficile de comprendre la superstition qui empêche les auteurs, les éditeurs, de mettre leurs textes sur la Toile. Croient-ils qu'ils en vendront moins ? C'est tout le contraire, si l'on pense à des essais qui de toute façon ne se diffuseraient qu'à quelques centaines d'exemplaires. Les moteurs de recherche maintiennent le texte en vie, le signalent aux curieux, aux chercheurs.

J'ai participé à une réunion chez les « Verts » où l'on parlait de la diffusion de textes gratuits sur la Toile. Tous disaient qu'il fallait l'encourager ; certains soutenaient même que le caractère individuel de l'écriture était une fiction et idéalisaient la pratique médiévale de l'anonymat des artistes. Plusieurs d'entre eux avaient cependant publié des livres et des articles. « Et toi, leur ai-je demandé, as-tu mis tes travaux sur la Toile ? Non ? Qu'est-ce qui t'en empêche ? » Les réponses, embarrassées, furent du genre « Je ne m'y connais pas assez en informatique ».

Allons donc ! quiconque sait taper un texte peut utiliser FrontPage, ou l'un de ces logiciels si commodes dont on se sert pour publier des blogs.

Revenons sur le caractère individuel de l'écriture. Si l'auteur est porteur d'un flux qui le traverse plutôt que son créateur véritable, il ne faut pourtant pas gommer toute référence à son individualité. Un auteur ne se comprend bien que si l'on prend le temps d'entrer dans son langage, son vocabulaire, son orientation ; pour interpréter ses textes, il faut se placer dans son monde mental.

Le dialogue entre le lecteur et l'auteur est le but principal et même unique de l'écriture, même s'il se déroule pratiquement toujours à l'insu de l'auteur¹⁸.

Tous ceux - consultants, journalistes, écrivains, philosophes, chercheurs : penseurs en un mot - qui mettent en forme des idées et créent ainsi du contenu devraient le mettre à disposition sur la Toile en utilisant la *GNU Free Documentation License*. Les éditeurs devraient mettre sur leur site Web, en libre accès, le texte intégral et original des textes fondamentaux d'une discipline. Cela leur ferait honneur ainsi qu'aux auteurs. Les éditeurs de revues de faible tirage devraient mettre l'intégralité des articles sur la Toile : ce lectorat démultipliera leur influence et celle-ci fera finalement croître le nombre de leurs abonnés, car il est plus agréable de lire sur papier que sur l'écran et l'impression coûte plus cher qu'on ne le pense¹⁹.

18. « La lecture de tous les bons livres est comme une conversation avec les plus honnêtes gens des siècles passés » (René Descartes, *Discours de la Méthode* [46]). Cependant Marcel Proust, pour qui la relation avec autrui était douloureuse, a dit son désaccord avec Descartes dans sa préface de 1905 à la traduction de *Sésame et les Lys* de Ruskin : il trouvait dans la lecture non pas un auteur avec qui converser, mais une source d'émotions intimes : c'est une bonne illustration de la profondeur à laquelle s'enracine la diversité des points de vue.

19. Une cartouche pour imprimante à laser coûte de l'ordre de 100 € et permet d'imprimer de l'ordre de 2 500 pages ; une ramette de 500 feuilles coûte de l'ordre de 5 €. L'impression d'un livre de 500 pages reviendra donc à 25 € sans tenir compte de l'amortissement de l'imprimante elle-même, ce qui équivaut au prix du livre.

10.4 Annexe : Réseaux et protocoles

Dans un établissement d'une entreprise, les micro-ordinateurs sont connectés à un réseau (le « réseau local de PC » ou RLPC (en anglais LAN pour *Local Area Network*). Ces ordinateurs peuvent sur le RLPC communiquer entre eux, avec des serveurs de mémoire et de traitement, des imprimantes et des routeurs (serveurs de communication) leur permettant d'accéder au reste du monde.

Pour faire communiquer des ordinateurs, on peut imaginer une commutation de circuit comme pour le téléphone : la communication entre deux ordinateurs aurait lieu sur un circuit établi et réservé à la demande. Des raisons économiques militent cependant pour d'autres choix qui ont donné naissance à des réseaux comme « Ethernet » ou « Token Ring ». Leur principe est que tous les ordinateurs émettent leurs trames sur un même canal de transmission, et que les destinataires reconnaissent les trames qui les concernent.

Les ordinateurs qui communiquent sur un RLPC sont alors dans la même situation que des personnes placées dans une même pièce et entretenant plusieurs conversations à la fois. Les protocoles se distinguent par la façon dont ils règlent les problèmes que cela pose : comment distribuer le droit à la parole ? Comment faire si deux personnes parlent ensemble, ce qui empêche de comprendre ce qu'elles disent ? Peuvent-elles continuer à parler lorsque cette « collision » se produit, ou doivent-elles se taire dès qu'elles s'aperçoivent du problème ? Les performances permises par chaque protocole dépendent des règles adoptées.

10.4.1 Largeur de bande, débit utile et « overhead »

Le canal de transmission se caractérise par sa « largeur de bande²⁰ », ou mieux son débit, exprimé en bits ou en octets par seconde²¹. Sur un réseau Ethernet, le débit le plus courant est de 10 Mbit/s mais des versions plus rapides existent (voir page 360).

Selon le protocole, une trame peut mesurer de 72 à 1518 octets. La durée d'émission de la trame est fonction inverse du débit sur le réseau. Le « débit utile » est égal au nombre de bits que le réseau peut transmettre par seconde sans collision : c'est une fraction du débit total. Pour calculer la durée d'un transfert de fichier il faut soustraire à ce débit utile l'« overhead » provoqué par le découpage du fichier en trames ainsi que par l'adjonction au contenu

20. L'expression « largeur de bande » concerne les circuits analogiques, dont la capacité de transport se mesure selon la largeur de l'intervalle de fréquences qu'ils sont capables de transporter sans déformation (plus précisément, avec une déformation qui reste inférieure à un seuil conventionnel). On l'utilise aussi, par contiguïté linguistique, pour désigner le débit sur un réseau numérique. C'est selon une approximation analogue que l'on appelle « modems » les adaptateurs d'un réseau numérique, par exemple sur les liaisons ADSL.

21. Avant 1975 « byte » pouvait désigner un paquet de bits de taille quelconque : Knuth ([105], vol. 1 p. 125) utilise ainsi pour l'ordinateur MIX un « byte » de six bits. Vers 1975 « byte » a pris le sens d'« octet » qu'il a aujourd'hui, soit huit bits. Il ne faut pas confondre « byte » et « bit » !

utile des adresses et contrôles. De tout cela résulte que le débit utilisable pour transporter du contenu n'est qu'une fraction du débit physique offert par le réseau. Cette fraction sera d'autant plus élevée que le protocole est plus performant.

On peut analyser les performances permises par chaque protocole en partant des lois probabilistes des files d'attente. Appelons « durée de trame » la durée (fonction inverse du débit) de l'émission d'une trame par un ordinateur, notons G le nombre moyen de trames émises pendant une durée de trame. Supposons que le nombre des trames émises pendant une durée de trames obéit à la loi de Poisson, qui rend compte du processus d'arrivée dans une file d'attente. La probabilité que le nombre des trames émises pendant une durée de trames soit égal à k est alors :

$$P(k) = \frac{G^k e^{-G}}{k!}$$

(*Nota Bene* : si l'on se rappelle que $e^G = \sum_{k=0}^{\infty} \frac{G^k}{k!}$, on vérifie aisément que $\sum_{k=0}^{\infty} P(k) = 1$ et que $\sum_{k=0}^{\infty} kP(k) = G$: le nombre moyen des trames émises pendant une durée de trame est donc bien égal à G).

Nous présenterons ci-dessous les deux protocoles de transmission sur réseau local qui ont été historiquement les premiers : « Aloha » et « Aloha discrétisé ». Étant simples, ils ont l'avantage de se représenter facilement sous forme mathématique ; la démarche qui a permis de passer du premier au second donne un bon exemple du type de raisonnement que font les concepteurs de protocoles. Nous présenterons ensuite les protocoles qui leur ont succédé et qui sont utilisés aujourd'hui.

10.4.2 Aloha

Le premier protocole de réseau local, nommé « Aloha », a été mis au point en 1970 par Norman Abramson [2] à l'université d'Hawaï. Il voulait assurer la communication entre des établissements de l'université situés sur des îles éloignées les unes des autres.

Le principe de ce protocole est que tous les ordinateurs émettent en même temps, reçoivent en même temps, donc communiquent en même temps sur la même bande de fréquence. Il s'agit d'une conversation à plusieurs, principe opposé à celui de la commutation de circuit qui procède en allouant à chaque conversation des ressources cloisonnées les unes par rapport aux autres.

L'émetteur découpe le message en trames comportant l'adresse du destinataire et un numéro d'ordre. Les ordinateurs reçoivent toutes les trames émises sur le réseau et trient celles qui leur sont destinées en lisant les adresses. Le destinataire reconstitue le message en ouvrant les trames ainsi triées pour en extraire le contenu et le ranger dans l'ordre après celui des trames précédentes.

Cependant si deux ordinateurs émettent une trame en même temps, il y a collision : le signal émis dans la bande de fréquence est incompréhensible. Il faut alors réémettre.

Notons τ la durée de trame. Pour qu'une trame ne provoque pas de collision, il faut qu'aucun autre ordinateur ne commence à émettre pendant une durée égale à 2τ (figure 10.5).

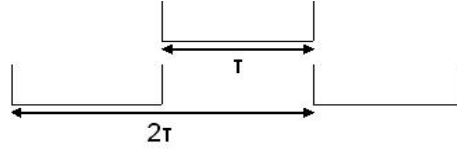


FIG. 10.5 – Succession des trames dans Aloha

En supposant que le nombre d'ordinateurs soit grand, la probabilité pour qu'il n'y ait pas de trame émise pendant la durée τ est $P(0)$. Or $P(0) = e^{-G}$. La probabilité pour que deux événements indépendants se produisent étant égale au produit de leurs probabilités, la probabilité pour qu'il n'y ait pas de trame émise pendant deux durées τ successives est $P^2(0) = e^{-2G}$.

Notons S le nombre moyen de trames « utiles » (c'est-à-dire émises sans collision) pendant une durée de trame. Pour qu'il n'y ait pas de collision, il faut que pendant deux durées de trame un seul ordinateur émette une trame, les autres restant silencieux. Comme en moyenne G trames sont émises par durée de trame, $S = Ge^{-2G}$.

La forme de cette fonction est indiquée par la figure 10.6. S atteint son maximum, égal à 0,18, pour $G = 0,5$.

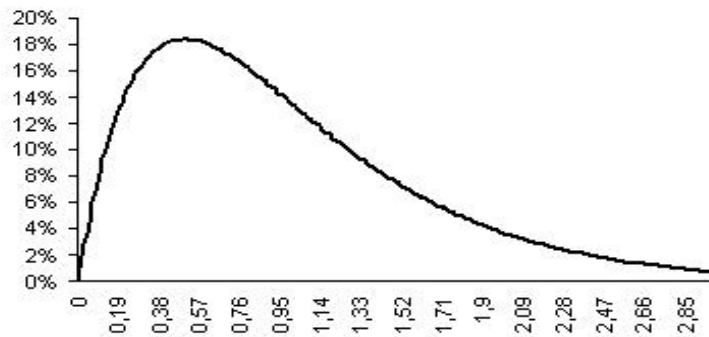


FIG. 10.6 – Rendement du protocole Aloha

Cette fonction mérite un examen attentif. Lorsque G est très petit, c'est-à-dire lorsque le nombre de trames émises par durée de trame est très faible, S est égal à G : le réseau n'étant pas encombré, les trames passent toutes sans collision. Si G croît, la probabilité de collision augmente: S devient donc inférieur à G .

Le maximum de S est atteint lorsque $G = 0,5$, c'est-à-dire lorsque sur le réseau une trame est émise en moyenne pour deux durées de trame. Alors

$S = 0,18$: en moyenne seules 36 % ($= 0,18 / 0,50$) des trames émises passent sans collision. Le nombre de trames utiles par durée de trame étant 0,18, le réseau peut véhiculer des données pendant 18 % de son temps d'utilisation. Ce taux est un *maximum* : le débit disponible pour le transport des données est égal au plus à 18 % du débit physique du réseau.

Si G croît au delà de 0,5, le nombre de collisions croît encore et le nombre de trames utiles décroît. Pour des valeurs importantes de G , S sera très faible : le nombre de collisions est tellement grand que le nombre des trames utiles devient très petit²².

Aloha discrétisé

En 1972, Roberts (Roberts [174]) mit au point une version perfectionnée du protocole Aloha : une horloge installée sur le réseau émet un signal à la fin de chaque durée de trame. Un ordinateur n'aura le droit d'émettre qu'au reçu du signal de l'horloge : au lieu d'émettre une trame dès qu'il en a envie, il doit donc attendre le prochain signal. Une collision se produira si deux ordinateurs ont eu envie d'émettre pendant une même durée de trame, car ils émettront ensemble au reçu du signal d'horloge (figure 10.7).

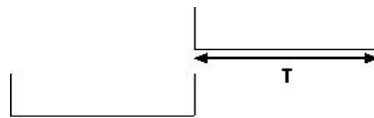


FIG. 10.7 – Succession des trames dans Aloha discrétisé

L'astuce de ce protocole, c'est de diminuer la durée du silence nécessaire pour éviter la collision. Cette durée était de 2τ avec Aloha, elle devient de τ seulement avec Aloha « discrétisé ». Ce perfectionnement a cependant un coût : il faut installer une horloge sur le réseau et mettre sur chaque ordinateur le dispositif lui interdisant d'émettre si ce n'est au reçu du signal de l'horloge.

On trouve alors :

$$S = Ge^{-G}$$

Le maximum de S est atteint pour $G = 1$ et il vaut 0,37. Aloha discrétisé est ainsi deux fois plus efficace qu'Aloha, puisqu'il permet d'utiliser jusqu'à 37 % du débit physique du réseau (figure 10.8).

10.4.3 Carrier Sense Multiple Access

Avec Aloha, l'ordinateur émet une trame dès qu'il en ressent le besoin. Or il est possible d'éviter certaines collisions si l'on fait en sorte que chaque

22. Certains trouvent surprenant que S ne devienne pas nul lorsque $G > 1$, puisque alors il y a plus d'une trame émise par durée de trame : c'est que G est un nombre moyen, et donc pendant certaines durées de trame il peut y avoir moins d'une trame émise même si $G > 1$.

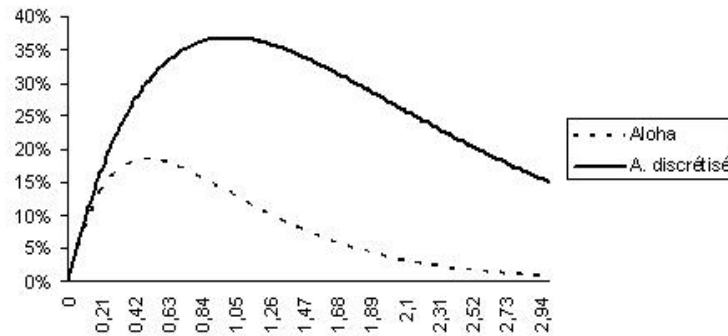


FIG. 10.8 – Comparaison des rendements d'Aloha et d'Aloha discrétisé

ordinateur écoute ce qui se passe sur le réseau avant d'émettre, et évite d'émettre lorsque le réseau est occupé : c'est comme si, dans une conversation à plusieurs, la règle était que tant que quelqu'un parle les autres se taisent. On distingue plusieurs types de protocoles CSMA (1975), tous plus efficaces que les protocoles Aloha :

CSMA persistant : l'écoute est continue et l'ordinateur émet dès que le réseau est disponible. Il y aura toutefois collision si deux ordinateurs ont attendu ensemble que le réseau soit disponible, parce qu'ils émettront chacun leur trame en même temps.

CSMA non persistant : l'ordinateur qui a trouvé le réseau occupé reprend son écoute après un délai aléatoire. Ceci permet de corriger le défaut précédent.

CSMA p-persistant : le temps est divisé en intervalles, comme avec « Aloha discrétisé ». Si un ordinateur veut émettre, il écoute pour savoir si le réseau est encombré. Il émet avec une probabilité p si le réseau est libre, et reporte l'émission à un intervalle suivant avec une probabilité $1 - p$. Le processus continue jusqu'à ce que la trame soit émise.

Carrier Sense Multiple Access with Collision Detection (CSMA-CD) : lorsqu'il y a collision, les dessins de trames se superposent dans l'espace hertzien et engendrent un signal incompréhensible pour les récepteurs. La détection de collision vise à limiter la durée de ce phénomène : l'émetteur écoute ce qui se passe sur le réseau, et dès qu'une collision se produit il interrompt immédiatement l'émission de la trame.

Un réseau sur lequel est utilisé un protocole CSMA-CD sera donc nécessairement à chaque instant dans l'une des trois situations suivantes : silence ; trame en cours de transmission (pendant une durée de trame) ; collision en cours (pendant une courte durée).

10.4.4 Ethernet

Le réseau Ethernet utilise un protocole CSMA-CD. Tout appareil connecté au réseau est appelé « nœud », qu'il s'agisse d'un poste de travail, d'un serveur (de fichier, d'impression, de communication etc.) ou d'une imprimante.

Chaque nœud est équipé d'une carte Ethernet installée sur l'un de ses slots d'extension. Chaque carte a une adresse sur le réseau. Les cartes Ethernet des nœuds sont raccordées chacune par un coupleur (« transceiver ») à une ligne de transmission appelée « bus²³ ». Quand un nœud veut envoyer des données à un autre nœud, sa carte écoute le bus pour s'assurer qu'aucun signal n'est en cours de transmission ; si le réseau est silencieux, elle émet sa trame sur le bus.

La trame comprend les adresses de l'émetteur et du destinataire, les données à transmettre, enfin des octets servant à la détection d'erreur. Chaque nœud examine l'adresse du destinataire. Si la trame ne lui est pas destinée, il l'ignore. Si la trame lui est destinée, il lit les données, vérifie qu'il n'y a pas eu d'erreur, et envoie un accusé de réception à l'émetteur qui peut alors envoyer la trame suivante.

Si deux nœuds émettent un message simultanément, la collision entre les trames provoque un signal d'interférence qui se propage sur le bus et qui est reconnu par les émetteurs. Le premier émetteur qui détecte la collision émet un signal indiquant celle-ci aux autres nœuds. Les transmissions sont alors arrêtées ; les nœuds qui veulent émettre doivent attendre une durée aléatoire avant de chercher à émettre de nouveau. Le processus se répète jusqu'à ce qu'un nœud puisse émettre une trame sans qu'il n'y ait de collision.

10.4.5 Protocoles sans collision

Pour améliorer le rendement du réseau, une autre possibilité est de faire en sorte qu'il ne puisse jamais se produire de collision. Chaque ordinateur peut à tour de rôle utiliser un droit à émettre : c'est comme si, dans une conversation à plusieurs, on faisait un tour de table. Le réseau doit alors supporter les délais dus à la circulation du droit à émettre.

Basic Bit-Map

On passe les ordinateurs en revue pendant une période de contention durant laquelle ceux qui souhaitent émettre sont notés. Puis chaque candidat à l'émission émet une trame, dans l'ordre des demandes.

23. Dans un ordinateur ou dans un réseau, un « bus » est un support de transmission sur lequel des signaux sont émis ou reçus par chacun des éléments qui lui sont raccordés. Seuls les nœuds auxquels les signaux sont adressés les traitent effectivement ; les autres les négligent. D'après Rosch [176], « bus » vient de la ressemblance avec les autobus que des passagers peuvent prendre ou quitter à chaque arrêt. Dans un réseau, le bus relie des ordinateurs ; à l'intérieur d'un ordinateur, le bus relie la carte mère avec les cartes insérées dans les slots d'extension (lecteurs de disques durs ou de CD-Roms, adaptateurs graphiques, cartes son etc.).

Les divers Ethernets

Le débit d'un réseau Ethernet est de 10 Mbit/s (Ethernet), 100 Mbit/s (Fast Ethernet) or 1000 Mbit/s (Gigabit Ethernet) (il s'agit du débit physique : le débit utile est plus bas en raison de l'*overhead* et des collisions).

L'Ethernet standard (10Base), nommé aussi « Thick Ethernet », utilise un câble coaxial qui peut parcourir 500 m sans répéteurs. Le raccordement d'un PC se fait en insérant un coupleur (« transceiver ») dans le câble coaxial ; le transceiver est lui-même raccordé à la carte Ethernet du PC par un câble.

Le « Thin Ethernet » (10Base2) utilise un coaxial plus fin et plus facile à utiliser, mais dont la portée est limitée à 200 m par segment. Les connecteurs sont en forme de T et le coupleur est incorporé à la carte du PC.

L'Ethernet sur paire torsadée (10BaseT) utilise le câblage téléphonique avec des connecteurs standard RJ-45 ; le câblage en étoile du réseau téléphonique suppose l'installation de *hubs*.

L'Ethernet rapide (« Fast Ethernet », ou 100BaseT) est analogue, mais utilise deux paires torsadées.

L'Ethernet sur fibre optique (10BaseF et 100BaseFX) est insensible à l'environnement électromagnétique et permet des segments de 2 km.

Token Ring et Token Bus

Supposons les ordinateurs rangés sur un réseau circulaire. Un bref message, le jeton (« token ») passe d'un ordinateur à l'autre. Le détenteur du jeton a le droit d'émettre une trame s'il en a besoin. Le réseau ne peut transporter qu'un message à la fois.

Les nœuds du réseau Token Ring sont connectés par un adaptateur à un bus en forme de boucle. Le jeton circule continuellement sur cette boucle. Chaque adaptateur lit le jeton à son passage. Le nœud qui veut émettre modifie le code du jeton pour signaler qu'il est utilisé et lui ajoute une trame comportant l'adresse du destinataire et quelques octets servant à la détection d'erreur. Pour éviter la dégradation du signal due à la transmission, chaque nœud comporte un répéteur qui régénère la trame au passage.

Chaque nœud examine le jeton lorsqu'il le reçoit et lit l'adresse du destinataire. Le nœud à qui la trame est destinée en fait une copie et la renvoie sur le réseau. La trame revient à l'émetteur qui la supprime et remet en circulation un jeton signalant que le réseau est libre.

Un protocole analogue peut être utilisé même si les ordinateurs ne sont pas rangés en cercle : il suffit que le jeton passe d'un ordinateur à l'autre selon un ordre régulier (*Token Bus*).

10.4.6 Domaines d'utilisation

Ethernet a été inventé par Robert Metcalfe et David Boggs chez Xerox en 1973. Sa version 1, mise au point par Xerox, Intel et Digital en 1980, a été utilisée par l'industrie aéronautique : Boeing a monté sur cette base son réseau TOP (« Technical and Office Protocol »).

Cependant sur un réseau Ethernet la durée d'attente d'une trame dépend d'une loi de probabilité. Le caractère aléatoire du délai d'émission n'est pas gênant dans l'industrie aéronautique ni dans les applications bureautiques où la production ne comporte pas de contraintes strictes de temps réel. Par contre elle est gênante dans l'industrie automobile où ce sont des automates qui communiquent et où les contraintes de temps réel sont donc très fortes.

General Motors avait besoin de majorer les délais de transmission des commandes vers ses robots. La technique du bus à jeton a été mise au point pour éviter les collisions et maîtriser les délais (réseau MAP, pour *Manufacturing Automation Protocol*). Enfin IBM a mis au point l'anneau à jeton Token Ring en 1985. Ces trois techniques ont été proposées aux organismes de normalisation qui n'ont pas voulu choisir entre elles puisqu'elles répondaient à des besoins différents. Ils les ont donc laissé cohabiter. On utilise donc aujourd'hui ces trois protocoles sur les RLPC :

- Ethernet, norme IEEE 802.3 de 1983 (origine : Xerox, puis DEC),
- Bus à jeton (Token Bus), norme IEEE 802.4 (origine : General Motors)
- Anneau à jeton (Token Ring), norme IEEE 802.5 (origine : IBM).

En pratique Ethernet s'impose sur les réseaux locaux à utilisation bureautique, Token Ring ou Token Bus étant utilisés pour des applications industrielles.

Chapitre 11

L'informatique de communication

L'informatique s'est historiquement développée autour de la saisie, du traitement et de la mise à disposition de données structurées. Mais avec le *groupware*, puis l'Intranet, se développe une « informatique de communication ». Il ne s'agit plus alors de traiter des données : les ordinateurs sont utilisés pour envoyer le courrier (messagerie), partager les agendas, diffuser la documentation etc.

L'informatique de communication utilise certes des données structurées (dans le carnet d'adresse de la messagerie, dans les masques de saisie ou de consultation de la documentation électronique) mais elles ne représentent qu'une faible partie du volume total de l'information manipulée. Les textes envoyés par message, ou consultables sur la Toile, sont écrits en langage naturel.

Dès lors se posent deux questions : quelles sont les potentialités de l'informatique de communication, comment la mettre au service de l'entreprise ? quelles doivent être les relations entre l'informatique de communication et l'« informatique de calcul », en désignant par ce dernier terme l'informatique consacrée au traitement et à la diffusion des données structurées ?

Pour faire le tour des potentialités de l'informatique de communication, nous examinerons d'abord les outils qu'elle comporte.

11.1 Les outils de communication

Messagerie

La messagerie permet entre les personnes une communication asynchrone qui bénéficie de la précision de l'écrit et de la souplesse du langage naturel. Elle comporte des pièges car son bon usage suppose un savoir-faire et même un savoir-vivre particuliers (voir page 387). Son insertion dans l'entreprise ne peut donc pas être immédiatement efficace : il faudra que les personnes apprennent à s'en servir, que son usage fasse l'objet d'une animation.

La messagerie est aussi un moyen de communication avec le monde extérieur (clients, partenaires, fournisseurs) : les relations commerciales en sont modifiées.

Annuaire électronique

Les adresses des utilisateurs de la messagerie sont stockées dans un annuaire électronique ; celui-ci peut être enrichi pour contenir également le numéro de téléphone, l'adresse postale, la photographie, la description des fonctions etc. de la personne.

On peut introduire dans l'annuaire assez d'informations pour alimenter le *profil* de la personne à partir duquel il sera possible de déduire ses habilitations : l'annuaire devient alors le pivot de la gestion des droits d'accès au système d'information.

Agenda partagé

L'agenda électronique permet aux personnes d'organiser leur emploi du temps ; répliqué sur un ordinateur de poche, il remplace avantageusement l'agenda papier. Mis en réseau, il facilite l'organisation des réunions et la prise de rendez vous par une assistante ou un collègue. Son bon usage suppose, comme celui de la messagerie, l'acquisition d'habitudes nouvelles.

Documentation électronique

La documentation électronique remplace avantageusement la documentation sur papier : elle permet de mettre à disposition sans délai des textes à jour, elle évite les oublis que peut comporter la diffusion du papier et l'empilage de notes qui modifient les versions antérieures, elle compense dans une certaine mesure l'inégalité qui existe entre les établissements de l'entreprise (directions régionales, direction générale) en ce qui concerne l'accès à l'information. Le moteur de recherche aide à trouver facilement la réponse à une question, les liens hypertexte permettent de relier entre eux les documents concernant des thèmes voisins.

Dissémination sélective

La dissémination sélective permet d'éviter cet excès de documentation qui fait que l'on dise « trop d'information tue l'information » : elle vise à fournir à chacun l'information dont il a besoin et uniquement celle-là.

Ce but est impossible à atteindre dans l'absolu, ne serait-ce que parce que les besoins d'une personne évoluent ; néanmoins on peut réduire le « bruit » de la documentation.

L'une des solutions les moins coûteuses consiste, en s'appuyant sur une segmentation des utilisateurs (voir 394), à diffuser des sélections documentaires adaptées chacune aux besoins d'un segment : des lettres périodiques, diffusées par la messagerie, peuvent par exemple indiquer les liens hypertexte vers des documents utiles en les associant à un bref commentaire. Ces lettres font connaître les nouveautés ou publient des revues thématiques.

Rédaction coopérative

Lorsque plusieurs personnes concourent à la production d'un même document, la gestion des versions successives est délicate. Il faut éviter la collision entre des corrections portant sur un même paragraphe (« concurrence »), ou les incohérences résultant de révisions mal coordonnées, ou

encore les ruptures de ton et de forme que suscite la diversité des rédacteurs.

Workflow

Le workflow permet de baliser le circuit des documents (tables d'adressage préprogrammées) et de normaliser leur présentation (masques de saisie). Il se situe à la charnière de l'informatique de communication et de l'informatique de calcul, car il se prête à la saisie et au traitement des données structurées et soulève donc des problèmes de cohérence. Il permet par ailleurs de calculer et diffuser automatiquement des indicateurs de volume et de délai : ainsi la qualité des processus devient visible.

La réalisation des workflows simples utilise des outils bureautiques, notamment la messagerie. Les grandes applications informatiques peuvent comporter elles aussi des workflows mais pour des raisons de performance elles utilisent des outils plus puissants que les outils bureautiques.

Les workflows sont adaptés aux procédures bien définies et balisées ; leur mise en œuvre est plus difficile dans un contexte non procédural. Les procédures bien définies sont nombreuses dans l'entreprise et suivent des parcours relativement simples (demandes de congé ou de mutation ; lettres de réclamation des clients ; préparation du budget annuel ; instruction des demandes de crédit, des contrats avec les fournisseurs et les partenaires etc.). Le workflow est efficace alors pour un coût modeste.

Gestion de la connaissance

Il est devenu banal de dire que la compétence est l'actif le plus précieux de l'entreprise. Certains des outils que nous venons d'énumérer (documentation électronique, rédaction coopérative, dissémination sélective etc.) facilitent la capitalisation et le partage de la connaissance dans l'entreprise. On peut aussi identifier les experts capables de dépanner oralement une situation bloquée, gérer l'annuaire qui permet de les atteindre en fonction de la question à traiter : la gestion de la connaissance mobilise ainsi non seulement la documentation électronique, mais les centres d'appel, les workflows, la messagerie etc.

Forum

Un forum (*news*) est une boîte aux lettres qui peut être consultée par plusieurs personnes (voire par tout le monde, selon la largeur de l'habilitation). Le forum apporte à la documentation professionnelle un complément parfois indispensable : il permet aux utilisateurs de poser des questions sur les domaines où la documentation est incomplète ou ambiguë, et aux experts d'apporter des réponses qui entourent la documentation d'un halo de commentaires. Les fournisseurs d'outils informatiques ont ainsi mis sur la Toile des forums destinés aux programmeurs et où l'on peut trouver des réponses aux questions que la documentation technique ne prévoit pas (comment contourner les bogues répertoriés, comment utiliser le produit dans des cas non classiques etc.).

Tout forum doit être *animé* : l'animateur purge les messages obsolètes ou non pertinents en regard du thème du forum, introduit dans le corps de la documentation le contenu des messages importants, sollicite les experts pour que toute question reçoive une réponse dans un délai décent. Un forum

mal animé devient bientôt une poubelle remplie de messages inutiles et les utilisateurs s'en détournent.

Outillage du dirigeant

Le dirigeant doit pouvoir trouver sur son PC les outils qui facilitent son travail : comme tout le monde, il utilisera la messagerie, l'agenda, la documentation électronique ; en outre il disposera d'un accès à des tableaux de bord, à des systèmes d'aide à la décision et de gestion de projet qui lui fournissent indicateurs et alarmes. Il peut se servir d'Excel pour faire des calculs sur les données ou pour les visualiser.

Réunion assistée par ordinateur

Il est commode, pour certaines réunions, de fournir à chaque participant un PC en réseau, un animateur assurant le suivi de l'ordre du jour. Les informations fournies par les participants, leurs suggestions, les résultats des votes sont affichés sur un écran, et à l'issue de la réunion le compte rendu et la liste des décisions prises sont publiés immédiatement. Cela ne peut toutefois réussir que si l'animateur est de bonne qualité, ce qui est étonnamment rare.

La réunion assistée par ordinateur peut s'envisager avec des participants dans une même pièce, ou avec des participants éloignés et reliés par le réseau. Elle convient bien aux réunions de suivi de chantier, ainsi que pour certaines réunions de remue-ménages (celles, par exemple, où il s'agit de choisir le nom d'un nouveau produit). Par contre elle ne convient pas à la négociation en face-à-face.

Exploitation d'enquête

Il est facile, pour réaliser une enquête interne à l'entreprise, de construire l'échantillon par tri dans l'annuaire puis d'envoyer à la liste des personnes enquêtées un message comportant un lien vers un questionnaire sur l'Intranet. On peut faire des relances automatiques vers celles qui n'ont pas répondu. Des programmes produisent les résultats de l'enquête sous forme de tris à plats, tris croisés et représentations graphiques qui facilitent et outillent l'interprétation.

Il est également possible de produire rapidement, pour un coût faible, des enquêtes auprès des clients (à condition toutefois que l'enquête soit réalisée dans des conditions telles que le taux de réponse soit élevé.

Accès externe à l'Intranet

Les cadres de l'entreprise prennent vite l'habitude d'utiliser l'informatique communicante : ils voudront pouvoir consulter la documentation professionnelle, et même accéder à leur poste de travail, depuis leur domicile, leur lieu de villégiature ou les locaux d'un client. De même, des partenaires auront besoin d'accéder à la documentation professionnelle en même temps qu'aux données. Par ailleurs certains des documents destinés à l'Intranet pourront utilement être placés sur le serveur Web public de l'entreprise. L'ouverture de l'informatique de communication à l'extérieur pose évidemment des problèmes de sécurité : on ne peut pas gérer efficacement la publication des documents sans gérer en même temps leur confidentialité et leur protection.

Sécurité

L'informatique communicante est d'autant plus vulnérable à la malveillance qu'elle est plus ouverte au monde extérieur, or cette ouverture est nécessaire dès que l'on veut pouvoir servir des utilisateurs nomades ou des partenaires.

Il ne faut pas prendre cette menace à la légère : les pirates pénètrent les systèmes d'information pour parasiter leurs ressources de mémoire et de puissance (ils installent leurs fichiers sur vos disques et font tourner vos processeurs), ou pire pour les saccager en détruisant des fichiers ou encore en commettant des indiscrétions. Toute grande entreprise est la cible de plusieurs attaques par jour, souvent maladroites mais parfois très affûtées (voir page 130). Il faut installer des pare-feux sur le réseau (*firewalls*), des programmes antivirus, hiérarchiser les serveurs de façon à isoler les données les plus sensibles etc. La sécurité est devenue pour les informaticiens une spécialité en relation avec l'administration de réseau.

11.2 Place dans le système d'information

L'informatique de communication a introduit dans le système d'information une dimension nouvelle. Elle facilite la communication de textes en langage naturel (et même la communication multimédia, avec l'outillage des plateaux téléphoniques et l'insertion dans les documents de liens vers des dessins, photographies, films ou enregistrements sonores). Elle se met ainsi au service de l'information connotée, alors que l'informatique de calcul s'était concentrée autour de l'information conceptuelle, des données structurées dont les référentiels fournissent la définition.

Ces deux formes d'informatique s'articulent : à l'occasion d'une communication, des données structurées sont consultées ou saisies ; d'autre part les outils de l'informatique communicante s'insèrent dans le processus de travail (à tel moment de la procédure, l'agent opérationnel transmet un dossier en pièce jointe à un message, amorce un workflow, consulte la documentation professionnelle etc.).

Cette articulation est un des enjeux du système d'information. Il dépasse la mission d'une direction de la communication, même quand celle-ci est chargée de la maîtrise d'ouvrage de l'Intranet : par exemple le déploiement d'un workflow suppose une réorganisation du processus de travail concerné.

L'informatique de communication obéit à des contraintes de « temps réel » moins exigeantes que celles de l'informatique de calcul. Le temps réel est en effet relatif à l'application considérée : il exige une fraction de seconde pour l'informatique transactionnelle mais seulement quelques minutes ou même beaucoup plus pour l'informatique de communication. Lorsqu'on envoie un message, on pense dans la plupart des cas qu'il sera lu au mieux dans la demi-journée : peu importe alors si sa transmission prend quelques dizaines de minutes. Le « temps réel » de mise à jour d'une documentation électronique sera le plus souvent de l'ordre de la journée. L'administration des réseaux tire parti de cette diversité des « temps réels » pour optimiser l'utilisation des ressources en définissant des priorités entre services, en

réservant des capacités aux services les plus exigeants, en gérant des files d'attentes etc.

* *

Malgré toutes ses promesses, l'informatique de communication se heurte à des difficultés qui ralentissent sa mise en place. Pour beaucoup d'informaticiens, dont l'expérience s'est formée dans l'informatique de calcul, « ce n'est pas de l'informatique » car elle ne traite pas les données structurées¹. Utiliser l'ordinateur pour faire de la messagerie, de la documentation électronique, des workflows, c'est selon eux encombrer les réseaux d'un débit de type nouveau et aléatoire : si les utilisateurs abusent des pièces jointes, si les workflows se multiplient, il ne restera plus de place pour les applications transactionnelles. Les personnes qui, au sein de la DSI, s'occupent de la « bureautique » sont parfois considérées comme des parias que leurs collègues qualifient dédaigneusement de « bidouilleurs micro ».

Pour les dirigeants, dont beaucoup ne sont pas utilisateurs du PC, les apports de l'informatique de communication sont douteux. Pourquoi, disent-ils, utiliser la messagerie alors que l'on peut envoyer des lettres ? pourquoi l'agenda électronique, alors que l'agenda sur papier est si commode ? la documentation électronique apporte des économies de temps et d'argent qu'ils perçoivent, mais le workflow leur semble être un artefact créé par des intellectuels : habitués à une informatique de calcul dont le formalisme est loin de celui des métiers, ils ne « réalisent » pas que l'informatique puisse outiller d'aussi près les procédures et la maîtrise de leur qualité.

Enfin, personne ou presque ne conçoit ce qu'apporte l'insertion de l'informatique de communication dans le système d'information. Le consultant qui l'évoque en réunion voit s'alourdir les paupières de certains auditeurs, d'autres entament une conversation ou se lèvent pour aller aux toilettes : ces signaux indiquent que la question n'est pas mûre et qu'il faut abréger l'exposé.

* *

Certains opposent, dans l'évolution du système d'information, une *logique de l'offre* et une *logique de l'usage*. Les tenants de l'usage prétendent que ce sont les utilisateurs qui décident, et tirent argument de détournements d'usage surprenants qui se produisent parfois lorsqu'on met en place un nouvel outil (ainsi, le développement de la messagerie sur le premier réseau d'ordinateurs a surpris, voir page 344). Les tenants de l'offre disent que c'est du côté d'Intel et de Microsoft qu'il faut chercher le ressort de l'évolution, car les utilisateurs ne peuvent utiliser que ce qu'on leur fournit.

Ces deux approches oublient le rôle des concepteurs, des organisateurs, des dirigeants, qui choisissent parmi les possibilités offertes et déterminent le rythme d'implantation de l'informatique communicante. Voici un scénario classique. L'entreprise a d'abord installé une messagerie, non sans quelques

1. Au milieu des années 1990, la maîtrise d'ouvrage d'une grande entreprise demandait à l'informatique d'installer une messagerie. Le directeur des études à la DSI répondit « la messagerie ne fait pas partie de mes douze premières priorités ».

erements : on a ouvert des boîtes aux lettres fonctionnelles pour les entités (direction, agence etc.), puis des boîtes personnelles en nombre limité. Il a fallu du temps pour voir qu'il était difficile de gérer une diffusion d'information empruntant la messagerie pour les personnes équipées, le papier pour les autres ; finalement, les réticences de l'informatique ont été surmontées et chacun a été doté d'une boîte personnelle.

On découvre alors que l'Intranet permet non seulement la messagerie, mais la documentation électronique. La direction de la communication est chargée de remplacer la documentation sur papier par des écrans. Mais il s'agit d'informatique et la « Dir Com » n'y connaît rien. D'ailleurs les informaticiens, qui avaient freiné la messagerie « par crainte d'un encombrement du réseau », sont encore plus réticents devant l'Intranet. Réunions, calculs de volumétrie, définition de normes, le temps passe. Enfin l'Intranet documentaire se met en place.

Alors on s'avise qu'il peut servir de vecteur à la communication interne et d'outil de travail non seulement à l'entreprise considérée dans son ensemble, mais aussi à chacune de ses diverses entités (directions centrales, directions régionales etc.) : une diversification s'opère, non sans désordre. Quelques pionniers envisagent d'utiliser l'Intranet pour des enquêtes, ou encore pour monter des workflows. L'informatique exprime sa préoccupation, les réunions se multiplient, expertises et chiffrages s'accumulent : les workflows ne risquent-ils pas en effet d'encombrer les réseaux ? etc.

Certains consultants disent que les choses ne peuvent pas avancer autrement, compte tenu des réticences des informaticiens, de l'inculture des dirigeants, de l'appréhension que suscitent les changements d'organisation. Oui, disent-ils, l'entreprise avance à reculons, sans conscience claire des enjeux, poussée par l'offre et en trébuchant sur de menus obstacles. Celui qui veut faire entrevoir à l'entreprise la logique de cette évolution ne sera pas entendu. Il faut procéder par petites réalisations exemplaires : elles seront imitées, et par contagion l'entreprise évoluera.

Ainsi on installe une petite messagerie, puis progressivement chacun aura sa boîte aux lettres ; ou bien on met en œuvre un petit workflow et son efficacité suscitera l'intérêt puis l'imitation. En suivant cette tactique on fait progresser l'entreprise alors même qu'elle n'a pas compris où elle allait.

Mais c'est bien là le problème. Si l'informatique communicante ne peut avancer qu'au coup par coup, il faudra bien qu'un jour l'entreprise perçoive cette évolution de ses procédures de travail et en tire les conséquences sur son organisation comme sur son positionnement. Il faudra bien qu'elle pivote pour avancer non plus à reculons mais en regardant devant elle, qu'elle sache évaluer les enjeux sans se laisser intimider par l'épouvantail du trafic². Il faudra que les concepteurs, les décideurs, se réveillent quand on parle d'informatique communicante et comprennent qu'il s'agit non d'une fantaisie pour intellectuels, non de caprices d'utilisateurs épris de nouveauté, mais d'une évolution importante qui mérite toute leur attention.

2. Les dépenses en télécoms représentant de l'ordre de 10 % du budget informatique, les économies devraient viser en priorité les 90 % restants.

La tactique de la « petite réalisation exemplaire » a d'ailleurs ses limites. J'ai installé chez certains clients des workflows efficaces dont les utilisateurs étaient très satisfaits, mais cela n'a pas suffi pour déclencher l'imitation. L'entreprise peut parcourir une certaine distance à reculons mais il faut une volonté lucide pour sauter en hauteur.

11.3 Sécurité

La cryptographie, c'est l'art que nous avons tous pratiqués au lycée lorsque nous adressions à nos copains des lettres écrites dans un alphabet secret. C'est aussi la branche des mathématiques qui sert à chiffrer et déchiffrer les messages que s'envoient des militaires ou des espions. L'un des sommets de son histoire est le déchiffrement de la machine allemande Enigma par Alan Turing (Hodges [84]).

Dans *Crypto* [114], Steven Levy a souligné avec une délectation visible le fait que ceux qui ont conçu les systèmes de cryptographie, n'étant pas les spécialistes les plus « pointus » de la discipline, l'ont abordé avec un regard neuf et que c'est cela qui leur a permis de trouver des solutions hérétiques, mais efficaces, auxquelles personne n'avait pensé auparavant.

11.3.1 Système à clé publique

Le système à clé publique a été découvert d'abord par James Ellis en 1969, puis par Whitfield Diffie en 1975 : nous verrons ci-dessous pourquoi il a été découvert deux fois. C'est un procédé hérétique : il suppose de publier une clé de codage alors que la tradition, appuyée sur une évidence apparente, voulait que la clé fût secrète.

Il faut, pour que ce système marche, disposer d'une fonction mathématique unidirectionnelle (*one-way function*) munie d'une porte arrière.

Fonction unidirectionnelle à porte arrière

Une fonction unidirectionnelle, c'est une fonction biunivoque $y = f(x)$ telle que, si l'on connaît une valeur y , il est pratiquement impossible de calculer la valeur x qui lui correspond - ou, en d'autres termes, d'inverser la fonction f .

La fonction unidirectionnelle est munie d'une « porte arrière » z s'il existe une fonction $x = g(y, z)$ qui permette, si l'on connaît z , de calculer facilement la valeur x qui correspond à y .

Supposons que Bob veuille recevoir des messages d'Alice, et qu'il souhaite que ces messages soient indéchiffrables pour Ève alors que celle-ci peut les intercepter.

Bob et Alice connaissent une fonction unidirectionnelle f . Bob fournit à Alice une « clé publique » c . f et c , pouvant être connus de tout le monde,

sont bien sûr connus d'Ève. Alice va coder son message M en utilisant l'algorithme f et la clé c ; ceci fournit un texte chiffré C qui ressemble à une suite de caractères tirés au hasard :

$$C = f(M, c)$$

Comme f est une fonction unidirectionnelle, Ève sera incapable de reconstituer le message même si elle connaît l'algorithme f , la clé publique c et le texte chiffré C .

Par contre Bob, lui, possède la « clé privée » z qui permet d'ouvrir la porte arrière de la fonction f . z est, elle, rigoureusement secrète et permet de déchiffrer le texte C en utilisant la fonction g :

$$M = g(C, c, z)$$

Pour que le procédé soit utilisable, il faut avoir défini les fonctions f et g ; cela demande une maîtrise des mathématiques que ni Ellis, ni Diffie ne possédaient. Le système à clé publique n'a donc d'abord été qu'une idée dont la faisabilité restait à démontrer. Les techniques de chiffrement ont été mises au point par des mathématiciens experts en théorie des nombres (Hardy et Wrigth [220]), une des branches les plus abstraites (mais aussi les plus intéressantes) des mathématiques (Singh³ [188]). La théorie des nombres est cruciale pour l'informatique, qui n'appréhende les nombres réels qu'à travers leur approximation par un sous-ensemble des nombres rationnels (ce qui, dit Donald Knuth, pose aux mathématiciens des problèmes parmi les plus difficiles; voir la note au bas de la page 83).

Certains algorithmes de chiffrement utilisent le fait que la factorisation (ou « mise en facteurs premiers ») du produit de deux grands nombres premiers demanderait aux ordinateurs les plus puissants un temps de calcul de plusieurs millions d'années, ce qui rend l'opération pratiquement impossible.

Le problème est ainsi résolu. Supposez en effet que la clé publique c soit le produit de deux grands nombres premiers, que la clé privée z soit l'un de ces deux nombres et que g comporte la factorisation de c : seul Bob, qui connaît z , pourra factoriser c et donc déchiffrer C .

Le système à clé publique devient universel si chacun publie sa clé publique dans un annuaire: pour envoyer un message chiffré à Bob, il suffit de trouver sa clé publique et de s'en servir pour chiffrer le message avant de le lui envoyer; seul Bob pourra le déchiffrer, puisqu'il est le seul qui possède la clé privée.

Il faut bien sûr que l'annuaire soit fidèle, car il se pourrait que la perfide Ève eût substitué sa propre clé publique à celle de Bob afin de pouvoir lire les messages destinés à celui-ci: des procédés ont été inventés pour traiter ce risque.

Si j'ai publié ma clé publique dans l'annuaire et que je reçois un message signé « Bob », qu'est-ce qui garantit que ce n'est pas Ève qui me l'a envoyé en imitant la signature de Bob? ici l'on rencontre une astuce d'une grande

3. Singh [187] est une introduction à la théorie des nombres pour le grand public.

beauté : le système peut servir non seulement à chiffrer les messages, mais aussi à authentifier la signature de l'expéditeur. Voici comment :

1) Bob chiffre sa signature avec sa propre clé privée ; puis il chiffre avec ma clé publique l'ensemble formé par le message M et sa signature déjà chiffrée. Cela donne le texte chiffré C .

2) Lorsque je reçois C , je trouve en utilisant ma clé privée le message M et la signature chiffrée de Bob ; je peux ensuite déchiffrer la signature en utilisant la clé publique de Bob. Cela me garantit que c'est Bob qui a envoyé ce message, puisque seul Bob peut chiffrer un texte en utilisant sa propre clé privée.

* *

Le chiffrement à clé publique apporte tout ce dont on a besoin pour communiquer en toute sécurité sur l'Internet, faire du commerce, protéger des travaux confidentiels etc. Mais ce système a beaucoup contrarié la National Security Agency (NSA) qui est chargée d'une part d'espionner tout le monde pour le compte des États-Unis (comme avec le système Echelon qu'elle a installé en Grande-Bretagne pour espionner l'Europe), d'autre part de protéger les secrets des États-Unis contre tout le monde.

Il y a donc eu des disputes, dans le style juridique propre aux Américains, entre tenants de la sécurité nationale et tenants de la confidentialité privée. Pendant un temps, les industriels du logiciel n'ont pas eu le droit d'exporter hors des États-Unis la version la plus efficace des algorithmes mais seulement une version qui, utilisant une clé plus courte, pouvait être déchiffrée dans un temps de calcul plus court (le temps de calcul double chaque fois que l'on ajoute un bit à la clé).

* *

Les recherches sur la cryptographie sont menées dans deux mondes : les chercheurs des services secrets, les chercheurs de l'université ou de l'industrie. Pourquoi le système à clé publique a-t-il été découvert deux fois ? parce que les travaux réalisés par la NSA comme par ses homologues étrangers sont secrets. Ellis travaillait chez l'homologue britannique de la NSA et sa découverte est restée longtemps secrète. Lorsque Diffie découvrit le procédé, il n'avait jamais entendu parler des travaux d'Ellis.

Les fonctions f et g sont de gros consommateurs de puissance informatique. Pour que l'algorithme puisse être utilisé sur l'Internet il fallait l'adapter au PC : ce fut le travail de Philip Zimmermann qui publia « Pretty Good Privacy » (PGP) en 1991, non sans encourir les foudres de la NSA. Le travail de Zimmermann a pris plusieurs années car pour mettre au point ce logiciel il a dû résoudre de délicats problèmes d'optimisation.

11.3.2 Identification, authentification et habilitation

On appelle « annuaire » ou « répertoire » une liste d'objets comportant pour chacun un identifiant et les valeurs de quelques attributs sélectionnés. Dans une entreprise, l'annuaire des personnes comporte classiquement leur

identifiant et leur état civil suivi de leurs diverses adresses (postale, téléphonique, de messagerie), de leurs mots de passe, ainsi que de leurs affectation, statut administratif et fonction.

On peut également construire un annuaire des matériels contenant leur identification et leur description (annuaire du réseau de PC, annuaire du mobilier etc.). L'annuaire du réseau de PC, qui décrit les PC eux-mêmes (versions des matériels et logiciels), serveurs et équipements du réseau, est utile pour la gestion de configuration des postes de travail.

Un annuaire est pour le système d'information un pivot d'échanges : on peut consulter l'annuaire des personnes pour trouver un numéro de téléphone, envoyer un message, composer une liste, établir des statistiques, authentifier l'utilisateur lors de sa connexion au système d'information, fournir le profil qui permettra de connaître ses droits (cf. ci-dessous).

L'annuaire peut être utilisé par une application de deux façons différentes : soit il est interrogé au coup par coup chaque fois que l'application a besoin de connaître la valeur d'un attribut (mode « pull »), et dans ce cas l'application doit connaître la liste des annuaires qu'elle utilise ; soit il est après chaque mise à jour répliqué dans la base de données de l'application utilisatrice (mode « push »), et dans ce cas chaque annuaire doit connaître la liste des applications qui l'utilisent et pouvoir les mettre à jour sans délai.

Le point le plus délicat de la gestion des annuaires réside dans leur délai de mise à jour. Si les modifications du monde réel (embauche d'une nouvelle personne, mutation d'un ancien, changement de la configuration d'un PC etc.) ne sont pas enregistrées dans l'annuaire sous un délai raisonnable en regard des exigences opérationnelles, le décalage entre l'annuaire et la réalité compromettra la qualité du système d'information.

Comme l'informatique équipe de plus en plus complètement les processus, le délai « raisonnable » devient court : on acceptera peut-être d'attendre une demi-journée avant de pouvoir installer une personne devant son poste de travail, mais pas plus. Il faut donc définir la procédure de mise à jour de l'annuaire, puis en contrôler l'exécution. Il sera souvent nécessaire de désigner la personne qui administrera l'annuaire et sera chargée de garantir sa qualité, en relation avec les personnes qui assurent les mises à jour et qui, elles, seront souvent dispersées dans le réseau, au plus près du terrain et donc du monde réel que l'annuaire doit refléter.

Un système d'information n'est pas un système ouvert dans lequel chacun pourrait fouiller à sa guise ou introduire des données selon sa fantaisie. L'accès aux données confidentielles (sur les personnes, les clients etc.) doit être réservé à des personnes habilitées, ainsi que la saisie ou la correction des données et que le lancement des traitements.

À chaque personne, qu'elle appartienne à l'entreprise ou à un partenaire, sont donc associées des *habilitations* spécifiques de consultation, écriture et traitement ; elles peuvent concerner la totalité d'un objet ou seulement une partie de celui-ci.

Lorsqu'une personne se connecte au système d'information, elle s'identifie (soit en fournissant son identifiant numérique, soit en utilisant un identifiant secondaire facile à retenir du type prenom.nom), et elle *authentifie*

son identification en fournissant un mot de passe (ou son empreinte digitale, l'image du fond de son œil etc.).

Les habilitations d'une personne sont fonction de certains de ses attributs que l'on regroupe sous le terme de « profil » (établissement de rattachement administratif, établissement où l'agent travaille, grade et fonction, équipe professionnelle etc.) : ainsi l'habilitation, bien qu'elle soit attribuée au coup par coup à une personne lors de sa connexion, n'est pas exactement individuelle car elle sera la même pour toutes les personnes qui ont le même profil.

Savoir comment et par qui le profil d'une personne est géré est l'une des articulations cruciales du système d'information. Si tous les attributs constituant le profil sont contenus dans l'annuaire, et si l'annuaire est en mesure de les tenir à jour de façon raisonnable, il sera possible lors de la connexion d'une personne de consulter l'annuaire pour y trouver son profil, puis d'en déduire ses habilitations.

Alors la tâche de l'annuaire est de tenir à jour les profils individuels des personnes, et la tâche du système d'information est seulement de tenir à jour la table qui définit les habilitations en fonction des profils : l'application n'aura pas à gérer des habilitations au niveau individuel.

Si par contre l'annuaire ne contenait pas tous les attributs qui constituent le profil, ou s'il n'était pas possible de le tenir à jour de façon raisonnable, chaque application devrait compléter les données de l'annuaire pour établir le profil de la personne. Elle comporterait donc elle-même, comme l'annuaire, des listes de personnes et d'attributs qu'elle devrait tenir à jour et administrer et cela alourdirait passablement sa gestion.

11.4 Les workflows

On appelle « workflow » l'outil informatique qui assure l'informatisation d'un processus : formulaire pour la saisie et la consultation, tables d'adressage, alarmes en cas de dépassement des délais, indicateurs statistiques de délai, de qualité et d'utilisation des ressources etc. Selon que le processus est plus ou moins compliqué, selon les contraintes de volumétrie et de performance, le workflow est plus ou moins complexe et certains petits workflows sont très simples. J'ai découvert le workflow en 1992 en lisant Peter Keen [99]. J'ai suivi de près la réalisation de plusieurs workflows chez mes clients :

- en 1995 Infotel, service d'assistance téléphonique à une force de vente. La réalisation, assez compliquée, a demandé de l'ordre de six mois et a coûté environ 300 000 € au client. Le service Infotel n'aurait jamais pu fonctionner correctement sans ce workflow (voir page 375) ;

- en 1996 un workflow pour les techniciens de maintenance des installations téléphoniques privées (PABX⁴). La réalisation, de complexité

4. « Private Automatic Branch Exchange » : les PABX sont des commutateurs privés installés dans les établissements ; ils utilisent des solutions techniques parfois différentes de celles retenues pour les commutateurs du réseau public.

moyenne, a demandé de l'ordre de quatre mois et a coûté 200 000 €. Là aussi, seul le recours au workflow a permis de répondre aux besoins ;

- en 1997, pour la préparation du budget informatique d'une grande entreprise, un workflow a permis de rassembler les descriptions de projets, synthétiser les demandes, préparer les arbitrages. Il a impliqué les MOAD et le contrôle de gestion. La réalisation, simple (une fois les spécifications faites), a demandé trois semaines de travail à un ingénieur expérimenté et a coûté 30 000 €. Le directeur général avait imposé des contraintes de délai intenable à la préparation du budget informatique, seul le workflow a permis de les respecter ;

- en 2000, un système de rédaction coopérative pour la documentation professionnelle. Ce projet a avancé grâce à un chef de projet frais émoulu de l'université et qui trouvait tout naturel de construire un workflow alors que ce ne serait pas venu à l'idée d'autres personnes. Les contributions, validations, approbations etc. obéissaient à un workflow qui automatisait l'adressage et permettait le contrôle des délais ;

- en 2001, un workflow pour la signature des conventions de partenariat (ces conventions suivaient auparavant des parcours compliqués, les délais de signature étaient aléatoires).

J'ai suivi d'autres workflows concernant des procédures de gestion mais leur mise en place est lente pour les raisons que j'évoquerai ci-dessous.

Le workflow permet d'améliorer les procédures par la maîtrise des délais et de la qualité. Il utilise des progiciels qui complètent la messagerie par des fonctions d'adressage automatique, traitement de formulaires, mesure des volumes et des délais. S'il est soumis à des exigences élevées en termes de volumétrie et de performance on utilisera des outils plus puissants, mais il faudra assurer leur intégration dans le système informatique. Le coût de la réalisation est dans la plupart des cas modeste une fois que l'on a spécifié ce que l'on voulait faire : la mise au point des spécifications demande souvent plus de travail que la réalisation technique.

Malgré son utilité et son coût modique, le workflow pénètre très lentement dans les entreprises :

- aux yeux de certains informaticiens la modicité du coût des petits workflows les classe parmi les « applications bureautiques » qu'ils ont du mal à prendre en considération, selon la règle « si c'est pas cher c'est pas sérieux ». J'ai vu un DSI choisir une solution à deux millions d'euros, qui demandait deux ans, de préférence à un workflow qui aurait coûté 100 000 € et demandé deux mois ;

- le workflow étonne les utilisateurs : lorsqu'on leur présente ses possibilités ils n'arrivent pas à y croire. Comme ils n'ont pas l'habitude de voir l'informatique sous ce jour ils ne « réalisent » pas ce qu'on leur dit. Ils ont l'impression que cela relève du rêve, que ce sont des considérations intellectuelles sans relation avec la « vraie vie » ;

- la bonne utilisation du workflow suppose qu'il soit administré par une personne dotée de pouvoirs suffisants. Comme cela chamboule les procédures il faut passer par une phase d'adaptation pénible : cette difficulté d'organisation constitue un obstacle plus important que la difficulté technique.

Il ne m'a été possible d'installer un workflow que quand il n'y avait aucun moyen de faire autrement : les responsables étaient aux abois, leur carrière était en jeu et les autres solutions comportaient soit des coûts, soit des délais incompatibles avec les engagements qu'ils avaient pris.

11.5 Études de cas

Pour apporter la chaleur du concret aux principes que nous venons d'évoquer, nous présenterons ici deux études de cas. Il s'agit de cas réels, mais nous avons masqué le nom des entreprises concernées.

11.5.1 Infotel

Après son lancement en 1989, Lotus Notes a joui d'un monopole de fait sur le marché du groupware : aucun autre logiciel n'apportait aux utilisateurs la même richesse de fonctionnalités ni une plate-forme permettant de programmer des applications spécifiques.

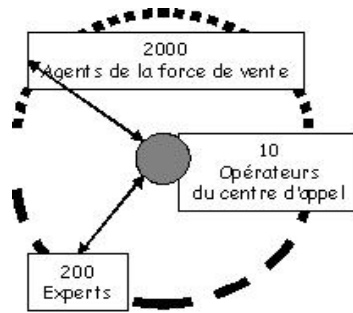
Par la suite Lotus Notes a eu des concurrents : Microsoft Exchange en 1996, puis surtout le *World Wide Web* lui-même. Le groupware, que certains jugeaient futuriste, est devenu courant et même banal au plan technique grâce à l'Internet et surtout à l'Intranet. Cependant il s'en faut de beaucoup que ses implications sur le plan de l'organisation des entreprises aient été tirées.

Ces implications sont illustrées par le service Infotel, mis en place en 1995 chez l'opérateur Euro Telecom. L'intérêt de ce cas réside moins dans son aspect technique, qui est daté, que dans ses effets sur l'organisation d'Euro Telecom et dans les leçons que l'on peut tirer des difficultés rencontrées.

* *

Au début des années 1990, Euro Telecom avait déployé une première ligne de 2 000 personnes dans ses agences commerciales. Cette entreprise fortement innovante lançait sur le marché un flot de nouveaux produits. Il fallait que les vendeurs fussent en mesure de répondre aux questions que posaient les clients, qu'elles portent sur la technique (utilisation du produit, interfaçage entre produits divers), les tarifs, le contrat ou sur tout autre sujet : « Tel client peut-il utiliser le réseau Datex-P de Deutsche Telekom pour ses relations avec la Pologne ? » ; « Un client localisé en Belgique propose d'acheter dix palettes de télécopieurs et paie avec un chèque tiré sur une banque luxembourgeoise ; que faire ? » ; « Tel client demande à renégocier sa ristourne de fin d'année, sachant qu'il prévoit telle croissance de sa consommation ; que lui répondre ? »

Les vendeurs recevaient une formation et une documentation lors du lancement de chaque nouveau produit et si cela ne suffisait pas ils pouvaient s'adresser à un centre d'appel. Le centre d'appel était lui-même en relation avec 200 experts disséminés sur l'ensemble du territoire et spécialisés chacun sur des questions techniques, commerciales ou juridiques (figure 11.1).

FIG. 11.1 – *Le centre d'appel au cœur du réseau d'expertise*

Selon les règles de la comptabilité analytique, les consultations d'experts étaient facturées 700 € à la première ligne. Cela lui coûtait cher car le centre d'appel se retournait vers les experts pour 90 % des questions.

Pour faire monter le taux des réponses directes, il fallait remplacer l'échange de télécopies entre le centre d'appel et les experts par des messages électroniques qui seraient placés ensuite dans une base documentaire indexée en texte intégral. Cette documentation faciliterait la capitalisation de l'expertise, les opératrices pouvant retrouver facilement la réponse aux questions les plus fréquentes et y répondre sans solliciter de nouveaux experts.

Mais si l'on retenait Lotus Notes pour cette documentation, on pourrait l'utiliser aussi pour monter un workflow des plus simples entre le centre d'appel et les experts, puis entre le centre d'appel et la force de vente elle-même. Il deviendrait ainsi possible de connaître la statistique des délais de réponse et, en associant quelques codes à chaque échange, de construire des indicateurs automatiques sur la qualité du système comme sur les types d'incidents qui lui étaient soumis.

Le projet évolua donc vers une définition plus complète, la base documentaire étant enrichie par un workflow. La réalisation technique fut rapide et relativement peu coûteuse, mais elle comporta des incidents instructifs.

Lors du lancement, quelques experts furent désignés comme « experts pilotes » et reliés au workflow, les autres continuant à recevoir des télécopies. Comme pour les opératrices du centre d'appel la messagerie était plus commode que la télécopie, les experts pilotes se trouvèrent submergés de travail. Ils en déduisirent qu'avec le nouveau système, ce serait l'esclavage : d'où révolte et réclamations.

Pour pallier cet inconvénient, il fallait que le travail à faire par l'opératrice fût le même, qu'il s'agisse d'un expert pilote ou non. Un petit programme permit de faire en sorte que l'envoi se fasse automatiquement, sous forme de message pour les experts pilotes et *via* le serveur de télécopie pour les autres.

Alors se produisit une catastrophe. Le serveur de télécopie associé à Lotus Notes n'avait pas été dimensionné pour envoyer les questions aux

experts. Surchargé, il s'est mis en panne ; et comme il était (pour des raisons de protocole de communication qu'il ne convient pas de détailler ici) solidaire du serveur Notes sur le réseau local du centre d'appel, la panne s'étendit à l'ensemble de l'application. Tout cela fut réparé assez rapidement mais non sans coût supplémentaire, car il fallut installer un deuxième serveur de télécopie au centre d'appel.

La maîtrise d'œuvre aurait pu prévoir cet incident si elle avait correctement anticipé les flux d'information : il existe en effet une analogie entre la physique de l'information (volumes, dimensionnement) et celle de l'hydraulique. Les flux d'information ne se compriment pas, ne se perdent pas ; ils se dirigent vers les canaux qui leur sont offerts et si ces canaux sont trop étroits (dimensionnement trop réduit, qu'il s'agisse des mémoires, des processeurs ou des circuits de transmission), ils cassent tout.

La maîtrise d'œuvre pensait avoir réglé l'ensemble des problèmes mais elle dut faire face à une nouvelle série d'incidents au moment de l'installation de l'équipement des experts. Les experts avaient des PC, mais qui n'étaient pas tous reliés à un réseau local (c'était en 1995) : or c'était indispensable pour qu'ils puissent utiliser l'application. Il fallait donc mettre les PC des experts à niveau. Dans un contexte d'équipement local sinistré la maîtrise d'œuvre a eu des surprises : ainsi un PC était bien relié à sa prise sur le réseau local, mais les fils de cuivre correspondants étaient « en l'air » dans le sous-répartiteur et en fait le PC n'était pas connecté, ce dont personne ne s'était rendu compte parce que l'utilisateur s'était servi jusqu'alors uniquement d'applications qui tournaient sur son PC. Un autre PC utilisait une carte Numéris qui marchait bien ; on change de carte, plus rien ne marche : il fallut du temps pour découvrir que la première carte était alimentée par le PC mais que la seconde carte devait être alimentée par la prise Numéris, et que les câbles d'alimentation n'avaient pas été raccordés à la prise de ce PC.

Pour régler de tels problèmes, il faut savoir poser le diagnostic et intervenir rapidement. Ce travail requiert des compétences techniques élevées, une grande habileté manuelle, de l'expérience et un esprit vif, toutes qualités qui ont un prix que le client n'est pas toujours prêt à payer. Le discrédit qui s'attache au travail manuel dévalorise en effet, aux yeux de beaucoup de personnes, tout ce que peut faire quelqu'un qui utilise un tournevis, même s'il s'agit d'un expert hautement qualifié et si cela demande, en fait, beaucoup de réflexion : c'est stupide, mais c'est ainsi.

D'autres incidents étaient dus aux fournisseurs : celui du logiciel du serveur de télécopie a modifié son produit ; l'ancienne version fonctionnait sur un processeur 386 mais la nouvelle exigeait un 486. La maîtrise d'œuvre a dû dire au client qu'il fallait acheter un micro-ordinateur supplémentaire...

Lors du passage à la « vraie grandeur », les travaux nécessaires pour l'installation sur de nombreux sites ont dépassé les capacités physiques de la petite équipe de la maîtrise d'œuvre, qui a dû les sous-traiter. Un premier sous-traitant, choisi sur la recommandation du chef de projet client, estimait avoir fait son travail lorsqu'il avait réalisé une liste convenue d'interventions, et ne se sentait pas concerné si l'installation du client ne fonctionnait pas.

Le client rendait la maîtrise d'œuvre responsable des conséquences et il a été difficile d'obtenir que ce sous-traitant soit remplacé par une grande entreprise disposant de personnels compétents dans toutes les régions.

Lors de la mise en œuvre opérationnelle d'Infotel, Euro Telecom a désigné un responsable d'application d'un niveau insuffisant pour assurer les tâches d'administration (contrôle et gestion des droits d'accès et des répliques entre serveurs) et d'animation : Euro Telecom estime en effet que la responsabilité d'une application est une tâche modeste, et il est difficile de faire désigner des responsables ayant le bon niveau.

Les personnes qui supervisaient le travail des experts ont vite compris qu'avec ce nouveau système elles n'auraient plus rien à faire : dans ce cas comme dans bien d'autres, la mise en place d'un workflow permettait de réduire le nombre des niveaux hiérarchiques. Ces personnes ont demandé au directeur ce qu'elles allaient devenir. Ce fut un moment délicat : alors que les anomalies techniques n'étaient pas encore corrigées, voici qu'un problème d'organisation se posait à chaud... Si le directeur avait manqué de courage, il aurait pu conclure à l'échec du projet. Heureusement il n'en fit rien.

* *

Grâce à Infotel, le taux de réponse directe par les opératrices du centre d'appel est passé à 40 % et la qualité du service s'est améliorée pour la plus grande satisfaction des utilisateurs. L'opération était donc un succès. Cependant Infotel a eu des effets pervers qu'il a fallu gérer :

- comme son utilisation était devenue facile, certaines personnes de la première ligne l'ont utilisé de façon paresseuse en appelant le centre d'appel au lieu de consulter la documentation. Les questions posées sont parfois devenues élémentaires. Il a fallu lancer une action de communication sur le bon usage d'Infotel ;

- les personnes de la première ligne n'appréciaient pas de recevoir une réponse directement d'une opératrice : la réponse est plus rapide que si l'expert avait été consulté, mais elle inspire moins confiance. Il fallut que la compétence du centre d'appel soit reconnue pour que la première ligne accepte de considérer ses réponses comme fiables. Ici aussi, une opération de communication s'est avérée nécessaire.

La montée en compétence des opératrices du centre d'appel a posé des problèmes de classification nouveaux pour Euro Telecom : les personnes passées par cette fonction avaient acquis une vue d'ensemble et connaissaient les incidents qui peuvent survenir. Elles étaient aptes à assurer des responsabilités opérationnelles et les fonctions d'opérateur auraient donc dû naturellement déboucher sur une promotion, mais ce n'était pas conforme à la « grille » maison (voir page 568).

Les questions que pose une force de vente doivent être interprétées : certaines reflètent des lacunes de la formation, d'autres traduisent une évolution du marché et sont susceptibles d'une exploitation par le marketing. Il est utile pour chaque chef de ligne de produit de recevoir des comptes rendus sélectifs contenant quelques statistiques et quelques courbes, ainsi que des questions et réponses significatives en texte intégral. Si le schéma de ces

comptes rendus est simple, sélectionner une information pertinente pour les confectionner relève d'une démarche éditoriale qui suppose de connaître les besoins d'information des destinataires.

Les comptes rendus sont fondés sur une information objective, disponible en temps réel sans qu'il soit nécessaire de faire une enquête supplémentaire, les exploitations statistiques se faisant sur les données produites au fil de l'eau par Infotel. L'exploitation de la base documentaire a également permis de fournir des comptes rendus aux directeurs régionaux. Il s'agit alors non seulement de diffuser des informations sur chaque produit, chaque région, mais aussi de tirer les leçons des comparaisons entre divers produits, diverses régions. Les éditions se démultiplient ainsi « sur mesure ».

Ainsi, à partir d'une demande initiale simple (accroître le taux des réponses directes par le centre d'appel) et pour un coût total relativement modeste (500 000 €, y compris le coût de la mise à niveau des PC des experts), la mise en place de la base documentaire et du workflow, puis de l'exploitation éditoriale, ont procuré une meilleure qualité de service, modifié l'organisation et la qualification des personnes, enfin permis une diversification des comptes rendus qui n'avaient pas été envisagée initialement.

11.5.2 Histoire d'un tableau de bord

Elseneur est une grande entreprise de services. Les tableaux de bord mis à la disposition de ses dirigeants au milieu des années 1990 présentaient des défauts manifestes : les données commerciales étaient biaisées (le chiffre d'affaires du dernier mois était sous-estimé d'environ 3 %), la présentation des séries chronologiques était fallacieuse : au lieu d'utiliser la correction des variations saisonnières (CVS), qu'ils jugeaient « trop compliquée », les dirigeants réclamaient que l'on présentât les données sous la forme R/R (« réalisé sur réalisé », valeur du mois divisée par la valeur du mois correspondant de l'année précédente) ou R/P (« réalisé sur prévu », valeur du mois divisée par celle prévue lors de la préparation du budget).

L'utilisation de ces quotients, où se mêlaient deux conjonctures différentes, leur interdisait de percevoir l'évolution et retardait la détection des retournements. Des variantes alourdissaient les tableaux de bord sans apporter d'information supplémentaire (R/R et R/P étaient parfois appliqués non à la valeur du mois, mais au cumul depuis le début de l'année, à la somme glissante des douze derniers mois etc.). Les commentaires paraphrasaient les tableaux de nombres en égrenant des pourcentages et fournissaient peu d'explications (il aurait d'ailleurs été difficile d'interpréter de façon correcte des indicateurs fallacieux).

Chaque service de l'entreprise se faisant un devoir de publier un tableau de bord le président-directeur général en recevait chaque mois quelques dizaines, tous conçus selon les méthodes que nous venons de décrire (donc incompréhensibles et en outre mutuellement incohérents). Il les faisait classer sans les regarder. Lors des réunions de direction, les écarts entre statistiques obscurcissaient l'interprétation des faits et occasionnaient de pénibles discussions.

Pourquoi l'on préfère un estimateur biaisé

Le biais de 3 % dans la première évaluation publiée du chiffre d'affaires mensuel d'Elseneur résulte du *principe de prudence* cher aux comptables : vers le 15 du mois $m + 1$, date de la première publication du chiffre d'affaires du mois m , certaines données ne sont pas encore parvenues. Si une région n'a fourni aucune information, on suppose son chiffre d'affaires nul ; si elle n'a fourni d'information que sur les quantités vendues, on les suppose vendues avec la plus forte ristourne.

L'évaluation définitive du chiffre d'affaires du mois m , publiée le 15 du mois $m + 2$, reposera par contre sur une information complète. Ainsi, et systématiquement, la deuxième évaluation sera supérieure à la première. Une personne a fait remarquer au responsable de ce calcul que le biais étant à peu près constant, il était possible le corriger. Il lui a répondu que s'il publiait des données sans biais, une fois sur deux la publication définitive réviserait à la baisse le chiffre d'affaires du mois précédent ; et il a conclu : « le directeur général admet les révisions à la hausse, mais à la première révision à la baisse je me ferai licencier ».

Le directeur général ignorait donc qu'un estimateur sans biais sera excessif une fois sur deux en moyenne ; il préférerait un estimateur biaisé au risque de fausser sa perception du marché.

Ces erreurs de méthode étaient ancrées dans les habitudes. Le contrôle de gestion était trop proche de la comptabilité pour pouvoir prendre le recul nécessaire au raisonnement économique. Il était impossible de corriger ces défauts en passant par les mêmes canaux de l'organisation.

La maîtrise d'ouvrage du système d'information et la direction de la stratégie ont proposé au président-directeur général d'établir un tableau de bord qui ferait apparaître l'évolution tendancielle des principaux indicateurs. Ce tableau de bord, baptisé par la suite modestement « bloc-notes tendanciel », sera succinct. Pour faire apparaître les tendances, chaque série sera corrigée des variations saisonnières et une extrapolation sera ajustée sur ses dernières valeurs. Les données seront accompagnées de commentaires signalant les retournements de tendance et expliquant les points aberrants (grèves, nouveaux produits, événements géopolitiques etc.).

Une page du bloc-notes contiendra une ou deux cellules, chaque cellule comportant :

- le graphique d'une série CVS (ou de quelques séries qu'il est utile de comparer), prolongé par une extrapolation de la tendance,
- un cadre contenant les nombres essentiels (niveau atteint, taux de croissance mensuel, taux de croissance tendanciel),
- un autre cadre contenant un bref commentaire.

Les données ainsi produites seront parfois différentes des données « officielles » que diffuse le contrôle de gestion, car elles sont corrigées des biais d'observation. Il convient donc de limiter leur diffusion pour éviter des mal-

Pourquoi il est difficile de construire un tableau de bord

Un tableau de bord, pour être efficace, ne doit fournir aux dirigeants que les indicateurs qui leur sont les plus utiles : un tableau de bord lourd ne sera pas lu. Pour sélectionner les indicateurs utiles, il faut savoir entrer dans le raisonnement stratégique, compétence rare. Il faut aussi traiter les données, les corriger, les interpréter pour en extraire l'information : cela suppose une expertise en statistique et en économie elle aussi rare. La démarche la plus fréquente consiste à produire des tableaux de bord lourds et à laisser le lecteur s'y retrouver : l'ambition est de décrire « complètement » la réalité de l'entreprise, but inaccessible dont on croit se rapprocher en publiant beaucoup d'indicateurs. Enfin, on suppose que l'« objectivité » consiste à transcrire telles quelles les données comptables en s'abstenant de toute estimation, alors même que ces données comportent des défauts évidents.

Sélectivité du tableau de bord

On est souvent tenté d'ajouter des indicateurs au tableau de bord pour accroître sa valeur.

Supposons cependant l'effort de lecture proportionnel au nombre des indicateurs publiés. Rangeons les indicateurs par ordre d'utilité décroissante et calculons l'utilité cumulée des n premiers indicateurs. La concavité de la courbe qui représente cette fonction de n est tournée vers le bas.

Si nous soustrayons de cette utilité cumulée la désutilité de la lecture, nous obtiendrons l'« utilité résultante » du tableau de bord ; dans le cas représenté dans la figure 11.2, elle est maximale pour cinq ou six indicateurs seulement.

Si on enrichit le tableau de bord, son utilité résultante décroîtra et elle deviendra négative à partir d'un certain seuil : alors le tableau de bord sera classé ou jeté sans être lu.

On ne peut évidemment en pratique quantifier ni l'utilité des indicateurs, ni la désutilité de la lecture ; il est bon cependant de garder à l'esprit ce raisonnement qualitatif simple car il aide à résister à la tentation d'enrichir le tableau de bord.

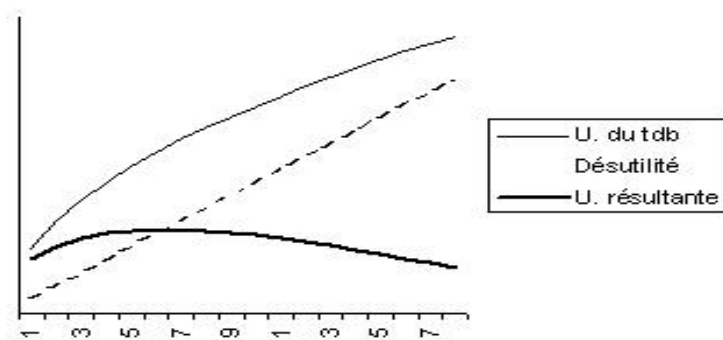


FIG. 11.2 – « Utilité résultante » d'un tableau de bord.

entendus : le bloc-notes sera réservé aux membres du comité de direction (quinze personnes).

Un « club du bloc-notes » rassemblera ses rédacteurs et les personnes qui leur procurent les données. Celles-ci auront le privilège de voir s'afficher les séries clés de l'entreprise et de participer à leur interprétation ; elles fourniront par la même occasion des commentaires utiles. L'existence de ce club permettra de désamorcer les réticences des « propriétaires » des données en leur donnant en échange une information privilégiée.

Le directeur général ayant été convaincu par une maquette qui montrait la pertinence de la méthode, il fut demandé à la direction de la stratégie de mettre le bloc-notes en place ; le travail sera réalisé par une petite équipe.

Une difficulté s'est présentée lors de l'élaboration du premier bloc-notes : pour construire une CVS, il faut disposer de données sur trois ans au moins ; or Elseneur venait d'absorber une autre entreprise. Il fallut reconstituer sur une durée suffisante, de façon rétroactive, l'entité fictive composée par la réunion d'Elseneur et de cette entreprise.

Le « club du bloc-notes » a été mis en place immédiatement. Les participants sont assidus. Le club est un élément clé du dispositif : les réunions intéressent les participants et fournissent des commentaires indispensables. Si l'entreprise n'avait pas eu de bonnes méthodes en matière d'indicateurs et de tableau de bord, elle disposait d'experts qui connaissaient parfaitement les données et étaient en mesure de les commenter. Les réticences à fournir les données étaient effacées.

Le contrôle de gestion, invité aux réunions du club, préféra d'abord ne pas y participer sans doute pour ne pas sembler cautionner cette initiative.

Le bloc-notes, publié vers le 25 du mois $m + 1$, comporte une partie fixe (séries régulièrement suivies) et une partie variable qui éclaire un problème conjoncturel jugé particulièrement important. Il se peut qu'un indicateur temporaire devienne permanent s'il répond à une priorité durable.

Les premières éditions du bloc-notes ont rencontré un vif succès auprès du comité de direction. Quelques exemples d'actualité ont permis de prouver

l'efficacité du concept et de la CVS : ainsi le bloc-notes a, le premier, permis de mettre en évidence les effets de la crise financière asiatique de 1998 sur Elseneur alors que les indicateurs du type R/R ne les faisaient pas apparaître.

Les premières éditions ne couvraient pas tout le domaine cible car la constitution du stock des séries CVS a été progressive. Puis le domaine cible lui-même a évolué pour s'adapter aux priorités stratégiques. En outre la partie variable a été enrichie d'un éditorial, petite étude très appréciée sur le sujet important du mois.

Dans sa première définition le bloc-notes devait fournir les grandes rubriques permettant un calcul de rentabilité et un suivi des parts de marché. Mais la stratégie de l'entreprise s'articule désormais selon quatre dimensions entre lesquelles le président-directeur général s'efforce de maintenir l'équilibre :

- 1) rapport qualité / prix du produit ;
- 2) avantages concurrentiels ;
- 3) gestion sociale et cognitive ;
- 4) fonction de production et création de cash-flow.

Ces quatre dimensions ont déterminé la présentation du bloc-notes. La gestion sociale et cognitive est le point le plus délicat : si le suivi social s'appuie sur un sondage hebdomadaire, Elseneur ne possède pas d'indicateur sur l'évolution de son capital intellectuel.

La philosophie que reflète ainsi le bloc-notes a toutes les chances d'être pérenne quels que soient les dirigeants futurs de l'entreprise.

* *

Le bloc-notes est essentiellement descriptif même si les commentaires apportent des éléments d'explication. Or on peut éprouver le besoin de modéliser des schémas de causalité, fussent-ils hypothétiques, pour les tester sur les données d'observation. Les tentatives d'application de l'économétrie aux séries chronologiques du bloc-notes ont cependant été décevantes. Le raisonnement n'utilise donc pas cette technique : il repose sur la comparaison visuelle de l'allure des séries, l'économétrie se faisant en quelque sorte « à l'œil » et non par le calcul.

Il ne faut d'ailleurs pas donner aux dirigeants des explications trop compliquées : la plupart d'entre eux n'ont plus fait de mathématiques après le lycée et il serait impossible de leur présenter un graphique en échelle semi-logarithmique. *A fortiori* ils ne sont pas prêts à entendre une expertise économétrique.

Il aurait été possible d'élargir la diffusion du bloc-notes mais le directeur général a jugé dangereux de mettre en circulation un document qui révèle les forces et faiblesses de l'entreprise : sa diffusion est donc restée limitée au comité de direction, ce qui lui a conféré d'ailleurs un grand prestige.

Il a été mis sur l'Intranet de l'entreprise sous un contrôle d'accès rigoureux. Les rares personnes habilitées peuvent trouver sur l'Intranet, outre le contenu de la dernière édition du bloc-notes, la série complète des éditoriaux

et l'historique des commentaires. On y trouve également les définitions et des précisions sur les périmètres couverts, ce qui facilite l'interprétation des données et limite les malentendus. Certains spécialistes ont accès chacun aux indicateurs concernant son domaine.

* *

Le bloc-notes est présenté au comité de direction par le DGA qui coiffe la direction de la stratégie. Ce directeur se fait « briefer » avant la réunion par le responsable du bloc-notes, et après celle-ci il lui rapporte les questions qui ont été posées.

Le bloc-notes est soigneusement et sérieusement examiné par le comité de direction : sa discussion occupe trois quarts d'heure dans une réunion de quatre heures. Il fournit aux dirigeants une base cohérente et les pertes de temps naguère occasionnées par les divergences entre statistiques ne se produisent plus. Il a permis de graver dans la mémoire des dirigeants le profil de certaines séries, ainsi que certaines proportions et ordres de grandeur essentiels.

Comme les calculs réalisés par l'équipe du bloc-notes n'ont jamais été mis en défaut un climat de confiance s'est créé. Les évaluations ne sont plus discutées. En cas d'évolution brusque la première question porte toutefois sur leur qualité, puis très vite le comité de direction passe à l'interprétation du phénomène.

Les commentaires sont appréciés : ils sécurisent le chiffrage en montrant que l'information a été analysée et contribuent à l'efficacité de la lecture. Même si certains dirigeants disent avoir tout compris avant qu'on ne leur explique, aucun d'entre eux ne réclame la suppression des commentaires.

L'extraction des tendances nourrit le commentaire : en fait, la tendance est l'information principale fournie par le bloc-notes. Elle a d'autant plus de valeur qu'elle a été validée par les experts du domaine. La succession des blocs-notes permet de signaler ses inflexions et retournements. Certains dirigeants prétendent voir la tendance sur le graphique sans qu'on la leur indique, mais ils ne pourraient sans doute pas détecter ses retournements.

Si la sélectivité du bloc-notes est une ascèse, celle-ci est appréciée. Il est accompagné d'annexes fournissant des ventilations par marché. Elles ne sont pas regardées par le comité de direction et ne sont là que pour apporter un éventuel complément d'information.

Le bloc-notes est une aide pour la décision, mais une aide indirecte. Il ne dicte pas la décision, qui suppose une réflexion et une concertation approfondies. Par contre, comme il permet aux dirigeants de partager une même connaissance des ordres de grandeur et des tendances, il leur fournit un cadre conceptuel commun qui facilite la préparation de la décision.

Par ailleurs les indicateurs révèlent parfois un problème de façon précoce et facilitent sa perception collective au sein du comité de direction, étape nécessaire (même si elle n'est pas suffisante) de la solution.

Au total, la pérennité du bloc-notes semble assurée. Son rattachement à la direction de la stratégie devrait être durable ; il pourrait être également rattaché au directeur général. Il faut en tout cas qu'il soit établi par une

direction généraliste. S'il était établi par le contrôle de gestion (et donc, dans le cas de cette entreprise, rattaché à la direction financière) il accorderait trop de place aux données financières. Toutefois le contrôle de gestion n'a jamais accepté que le tableau de bord soit établi par un autre service. Son opposition représente le principal risque politique.

Lorsque le bloc-notes a été lancé, le contrôleur de gestion a été invité aux réunions du club mais il n'a pas accepté d'y participer. Son successeur participe aux réunions. Il s'intéresse aux indicateurs, pose des questions, mais ne contribue pas à leur interprétation. Lorsqu'il est là, le climat de la réunion n'est d'ailleurs pas le même : le contrôle de gestion est craint parce qu'il détient un pouvoir sur les budgets, et en sa présence les participants se sentent moins libres de parler. Si sa participation aux réunions est en soi un fait positif, elles sont plus productives quand il est absent.

Le conflit s'est durci lorsque le bloc-notes a publié un indicateur mensuel du coût unitaire des produits. Le contrôle de gestion produisait des comptes d'exploitation par produit (CEP) en principe trimestriels, en fait souvent semestriels. L'équipe du bloc-notes a conçu un modèle pour estimer des coûts mensuels en s'appuyant sur les indicateurs et en se calant sur le CEP (l'économétrie est ici utilisée comme technique d'estimation). L'écart constaté après coup entre cette estimation mensuelle et la mesure trimestrielle que fournit le CEP est instructif.

La légitimité de ce travail a cependant été contestée par le contrôle de gestion et la décision a dû remonter à l'arbitrage du directeur financier. Le CEP a quelques défauts : comme il repose sur une comptabilité de trésorerie et non sur une comptabilité au fait générateur, il porte la trace des aléas des dates de facturation et de paiement. Cependant il était politiquement impossible pour le bloc-notes de s'écarter des coûts publiés par le CEP, du moins au niveau global et aux niveaux agrégés de la nomenclature des produits. Par contre aux niveaux détaillés les erreurs du CEP étaient par trop apparentes et l'équipe du bloc-notes a dû (et pu) s'affranchir de la contrainte du calage.

Le contrôle de gestion a finalement supprimé le CEP pour construire le modèle ACP (Analyse des Coûts de Production) alimenté par le progiciel SAP qui équipe Elseneur. Comme SAP ne fournit pas de coûts unitaires au niveau mensuel, ACP les évalue en multipliant les volumes par des coûts standards annuels : cela repose sur l'hypothèse, très grossière, que les coûts unitaires restent constants durant l'année. ACP fonctionne mal, ne satisfait personne, et le contrôle de gestion reste sourd aux avis des utilisateurs.

Il n'y a pas eu de recouvrement entre CEP et ACP : le bloc-notes a donc dû pendant un temps cesser de publier ses estimations de coût. L'équipe du bloc-notes est en train de mettre au point une méthode qui s'appuiera sur ACP et l'améliorera en répartissant au mieux la « poubelle » (quelques centaines de millions d'euros par mois) où s'accumulent les écarts entre les coûts constatés et l'estimation fondée sur les coûts standards. Le bloc-notes publiera ainsi des coûts détaillés différents de ceux que fournit ACP, et l'on peut s'attendre à un nouvel affrontement.

Enfin le contrôle de gestion a entrepris de mettre en place, dans la foulée de SAP, un « tableau de bord du directeur général » qui serait mis sur l'Intranet de l'entreprise. Il entend ainsi produire automatiquement 150 indicateurs mensuels. Ce projet coûteux, qui emploie une armée de consultants, tourne le dos à la méthode artisanale employée pour le bloc-notes. Il ferait retomber l'entreprise dans l'ornière du tableau de bord lourd, peu commenté, produit automatiquement et éventuellement mal vérifié. La cohabitation des deux tableaux de bord, si elle se produit un jour, suscitera la confusion dans l'esprit des dirigeants.

Il semble peu probable que le contrôle de gestion parvienne à construire un tableau de bord convenable. Pour le moment, le futur « tableau de bord du directeur général » ne publie que cinq indicateurs, diffusés à l'aide d'un modeste outil HTML qui demande 30 secondes pour afficher des images de courbes figées. Il est un peu choquant que ce projet ait pu consommer une telle part des ressources budgétaires, mais le contrôle de gestion central peut accéder à ces ressources-là plus aisément que les autres services.

11.6 Annexe : Bon usage de la messagerie

La messagerie n'est pas un moyen de communication banal : c'est un *média*. Comme il est relativement nouveau, il est souvent utilisé de façon maladroite. Voici des règles utiles :

Utiliser un ton aimable et poli

Introduite ou utilisée sans précautions, la messagerie suscite une inflation d'agressivité. Celui qui écrit un message croit souvent pouvoir ne pas surveiller son vocabulaire : il tape à la va-vite, envoie, et passe au message suivant. Mais parfois il aura utilisé une tournure familière, un ton désinvolte qu'il aurait mieux valu éviter. Le destinataire, froissé, rédigera une réponse savamment venimeuse. Le premier émetteur pensera que l'autre manque d'humour et voudra lui donner une leçon. C'est ainsi que le ton monte, et que des personnes qui s'entendaient bien auparavant ne se saluent plus lorsqu'elles se croisent.

Il faut en prévenir les utilisateurs novices : si un message ne demande pas le même cérémonial qu'une lettre, il est opportun d'y placer des signes de considération. Commencez par « Cher Monsieur », « Cher Ami » ou « Cher Jean-Claude », terminez par « Amitiés », « Meilleurs sentiments » ou « Cordialement », n'utilisez pas l'ironie ni l'invective envers le destinataire (elles sont permises envers les tiers, mais pensez à une autre règle que nous verrons plus loin : « ne rien écrire dont on puisse regretter la diffusion »).

Écrire en bon français

Celui qui reçoit le message préfère un texte simple, clair et bien écrit⁵. Les fautes d'orthographe et les solécismes sont signe de désinvolture. Il faut donc passer le correcteur d'orthographe avant d'envoyer un message, le relire pour en améliorer la formulation.

Cela prend du temps mais c'est un temps bien utilisé car les fautes de français provoquent des ambiguïtés. Ce qui est en question c'est donc, outre la politesse, la clarté de la communication. Le style de vos messages correspond à votre personnalité : les destinataires en perçoivent et en interprètent toutes les nuances.

Écrire des messages courts

Le destinataire vous saura gré de ne pas devoir utiliser l'ascenseur placé sur le côté de la fenêtre du message. Il faut donc que le message soit assez bref pour n'occuper qu'une hauteur de fenêtre. Ce n'est pas toujours possible, surtout lorsqu'on répond à une question délicate : cette règle tolère donc de fréquentes exceptions mais cela ne lui enlève rien.

Veiller à la compatibilité avec les logiciels du destinataire

Une des fonctions les plus commodes de la messagerie, c'est de faire du message le poisson pilote du transfert de fichiers, le bordereau d'envoi d'une pièce jointe. Mais le destinataire ne pourra ouvrir une pièce jointe que s'il dispose du logiciel qui le permet. Si vous avez la dernière version de Word ou de PowerPoint, les personnes équipées de versions plus anciennes

5. Il n'est pas toujours indispensable d'aller jusqu'à l'imparfait du subjonctif...

ne pourront pas ouvrir les documents que vous leur envoyez (d'où message de réclamation, nouvel envoi et perte de temps).

Notez les indications que donnent vos correspondants usuels et veillez à transcoder avant envoi les fichiers dans un format qu'ils puissent utiliser. Pour les correspondants occasionnels, si vous utilisez une version récente du logiciel, transcodez d'office dans la version antérieure. Si un correspondant usuel utilise la compression Zip, compressez les documents volumineux avant de les lui envoyer : le transfert sera plus performant.

Utiliser le moins possible les listes de diffusion

Certains croient leur parole d'une telle importance qu'ils en inondent les boîtes aux lettres de toute l'entreprise en utilisant systématiquement des listes de diffusion. Or il ne faut envoyer un message qu'aux personnes qu'il peut intéresser. Si vous abusez des listes, vos messages ne seront pas lus et votre réputation souffrira. Si l'un de vos collègues utilise les listes de façon indiscreète, concertez-vous avec les autres pour utiliser l'arme absolue : répondre tous en termes gentiment ironiques. Sa boîte aux lettres sera inondée et il cessera de vous importuner.

Dans certaines entreprises, les boîtes aux lettres sont encombrées de notes administratives à diffusion générale. Cette inondation décourage les utilisateurs et stérilise la messagerie. Les notes administratives seront de préférence placées sur un serveur Intranet et la direction générale diffusera une fois par mois un message récapitulatif décrivant brièvement leur contenu et fournissant des liens pour les consulter. Il faut donner à ce message récapitulatif une large diffusion ; les listes - c'est encore une exception à la règle mais elle se comprend aisément - conviennent bien pour informer plusieurs personnes de l'évolution d'un site Web. Il est poli d'indiquer aux destinataires comment procéder s'ils préfèrent que leur nom ne figure plus sur la liste.

Consulter sa boîte aux lettres au moins une fois par jour

Les personnes qui vous envoient un message savent que vous ne le lirez pas tout de suite (contrairement au téléphone, la messagerie est asynchrone) mais ils espèrent que vous le lirez dans les heures qui suivent. Il faut donc consulter votre boîte aux lettres au moins une fois par jour. Quand vous êtes en déplacement, vous pouvez consulter vos messages en utilisant un service (gratuit) de messagerie sur la Toile comme celui de Yahoo ou de Google.

Faire en sorte de ne pas recevoir plus que quelques dizaines de messages par jour

Si vous recevez plus de vingt à trente messages par jour, vous risquez de vous dégoûter de la messagerie et de ne plus la consulter. Il faut utiliser des filtres pour réduire le flux qui vous arrive : si quelqu'un vous envoie régulièrement des messages sans intérêt, détruisez les ou notez les comme *spam* ; classez dans des dossiers les messages diffusés par des sources d'expertise comme ZDNet ou Jesse Berst, vous les lirez quand vous en aurez le temps ; transférez vers votre collaborateur concerné les messages de ceux qui, respectant à l'excès la voie hiérarchique, ont cru devoir s'adresser à vous. Il suffit de paramétrer votre messagerie pour qu'elle exécute automatiquement ces diverses opérations.

Le travail de certaines personnes nécessite qu'elles traitent plusieurs centaines de messages par jour : la règle ci-dessus ne s'applique évidemment pas à elles.

Répondre aux messages reçus

Souvent celui qui a envoyé un message attend une réponse ou un accusé de réception. Ne remettez pas à plus tard la rédaction de la réponse - mais rédigez la avec soin, cf. plus haut.

Cette règle comporte des exceptions : il ne faut pas répondre aux messages qui n'ont pas besoin de réponse, et c'est la majorité des messages que nous recevons... à chaque message, il faut donc se poser la question : « dois-je répondre ou non ? », et agir en conséquence.

Il faut éviter de s'enfermer dans des échanges sans fin : « il m'a écrit, je lui réponds, il répond à ma réponse, je réponds à sa réponse » etc. Le plus avisé des deux est dans ce cas celui qui rompt le cercle en cessant de répondre.

Ne rien écrire dont on puisse regretter la diffusion

Le destinataire d'un message est propriétaire de celui-ci et peut donc le montrer à qui il veut. La personne à qui vous faites une confiance par messagerie peut la dévoiler, même si c'est plutôt indélicat.

N'abusez pas des plaisanteries sur les tiers, ne révélez pas crûment le fond de votre pensée, ne vous laissez pas aller : un de vos messages pourrait être imprimé et diffusé, ou encore transmis à la cible de vos railleries.

Diversifier les utilisations de la messagerie

L'entreprise peut gagner en efficacité en diversifiant les utilisations de la messagerie ; voici quelques exemples :

1) La messagerie est d'abord le « poisson pilote » du transfert de fichiers : vous pouvez mettre en pièce jointe un fichier Word, Excel, PowerPoint etc., ou encore la nouvelle version d'une application que le destinataire pourra installer sur son PC.

2) On peut classer les messages concernant un même sujet dans l'ordre de la conversation, faisant ainsi apparaître questions et réponses dans leur ordre naturel.

3) La messagerie fournit enfin leur plate-forme aux workflows qui équipent les processus les plus simples : le paramétrage des tables d'adressage, « timers », alarmes et compteurs, permet de contrôler la qualité des processus.

Veiller à ce que toute l'entreprise utilise la messagerie

Cette recommandation s'adresse non à l'utilisateur individuel, mais à l'entreprise. Si seule une moitié des personnes utilise la messagerie, il faudra deux circuits de documents : des messages pour ceux qui ouvrent leurs boîtes aux lettres, le papier pour les autres. La gestion des listes de diffusion en sera compliquée. Cela se produit lorsque l'entreprise n'attribue de boîte aux lettres qu'à quelques personnes, ou que certains salariés ne s'y « mettent pas », ou encore que les boîtes aux lettres sont remplies de messages inutiles et donc non consultées.

Certains dirigeants font gérer leur boîte aux lettres par leur assistante qui trie les messages, imprime les plus importants, et tape les réponses qu'ils

ont portées en annotation. Je conseille à ces dirigeants de gérer eux-mêmes leur boîte aux lettres mais il faut que l'assistante puisse elle aussi les lire, ainsi que les réponses, sinon elle ne connaîtra plus les affaires que suit son patron et ne pourra pas l'assister efficacement.

Il est préférable que les utilisateurs disposent tous de la même version des logiciels de base (traitement de texte, tableur, grapheur) car cela facilitera l'utilisation de la messagerie pour le transfert de fichiers. Il est préférable que ces versions des logiciels soient récentes, car cela facilitera la communication de l'entreprise avec le monde extérieur (si vous avez la dernière version de Word, vous pouvez ouvrir tous les documents Word que l'on vous envoie ; si vous n'avez qu'une version ancienne, vous ne pourrez pas ouvrir les documents composés avec les versions récentes).

Administrer la messagerie

Pour veiller à la qualité de ce média, l'entreprise doit nommer un « administrateur de la messagerie » outillé non pour lire les messages, mais pour mesurer le nombre de messages reçus chaque jour par chacun, la fréquence d'ouverture des BAL, celle des réponses, la longueur des messages, l'utilisation des listes de diffusion, l'encombrement du serveur, les délais de transmission etc. Il peut ainsi repérer les utilisateurs maladroits et leur indiquer, gentiment et poliment, le chemin vers la bonne pratique.

Quatrième partie

**L'informatique dans
l'organisation**

Chapitre 12

Acteurs, fonctions et rôles

12.1 Le marketing interne

On dit souvent qu'il faut « connaître l'utilisateur », « répondre à la demande des utilisateurs » etc. Ces phrases contiennent à la fois une vérité et un piège. Certes le système d'information est fait pour être utilisé : il doit donc répondre aux besoins des utilisateurs et sa définition doit s'appuyer sur une connaissance exacte de ce qu'ils font, de la diversité de leurs situations etc.

Mais l'utilisateur du système d'information, ce n'est pas l'individu affectif dont chacun de nous est un exemplaire particulier ; c'est l'« être humain organisé », personne dont la compétence professionnelle, la créativité, s'articulent à celle d'autres personnes pour constituer l'entreprise comme organisation de compétences.

Chaque utilisateur a ici deux faces : son individualité, son affectivité, sa culture, ses connaissances, bref ce que l'on désigne par le mot « personnalité » ; et la fonction qu'il remplit dans l'entreprise, son insertion dans le processus de travail. Ces deux faces communiquent à l'intérieur de la personne : la qualité du travail que celle-ci fournit est nourrie par sa culture, son imagination, ses connaissances, son caractère¹. Pourtant il faut les distinguer : de la personne l'entreprise ne retient en effet que la « compétence », c'est-à-dire l'aptitude à nourrir les processus de travail qu'il s'agisse de produire, organiser, contrôler ou concevoir.

Faire ainsi abstraction de la dimension affective de la personne peu sembler choquant mais c'est fondamentalement sain : il est à bien des égards préférable que l'entreprise ignore la vie privée, l'intimité et les opinions personnelles du salarié, car elles n'appartiennent qu'à lui et ne la concernent pas.

1. Dans le cas particulier de la programmation, l'écart de productivité entre un programmeur expérimenté et intelligent et un programmeur maladroit est non pas de deux ni de dix, mais plutôt de mille ou de dix mille.

Il faut par ailleurs distinguer les « besoins » de l'utilisateur de sa « demande ». Certes l'utilisateur est, tout comme le consommateur, *seul porteur authentique de ses besoins* : pour définir ce que doit faire le système d'information il faudra l'interroger et l'observer. Mais si la demande exprime le besoin, elle y mêle aussi l'idée que l'utilisateur se fait du possible et du normal ainsi que sa perception de l'offre disponible².

L'expression du besoin peut être alors déformée dans deux directions opposées : l'évolution technique étant rapide, la demande risque d'être trop timide et de se référer à un état de l'art dépassé ; ou bien au contraire, influencée par des annonces commerciales fallacieuses, elle transcrira des rêves de science-fiction.

Au total, *la demande n'est jamais la transcription fidèle et pertinente du besoin*³. Il faut la traduire pour remonter au besoin, tout comme l'on doit traduire les paroles d'une personne pour remonter à une intention qu'elle exprime dans son langage. Il faudra de plus, pour tenir compte d'exigences de coût et de qualité que l'utilisateur ne peut pas connaître (simplicité et solidité de l'architecture, sécurité du fonctionnement, pérennité des solutions, évolutivité etc.), ajouter à la prise en compte de ses besoins celle des contraintes techniques.

Répétons cependant que l'utilisateur est le seul porteur authentique de son besoin et que l'on ne peut connaître celui-ci qu'en l'écoutant et en l'observant. La situation du système d'information vis-à-vis de l'utilisateur est la même que celle de l'entreprise vis-à-vis de son client : l'offre ne peut apporter d'utilité, d'efficacité, que si elle se fonde sur la connaissance du client.

Or toute connaissance du client s'appuie sur un *classement* et suppose donc une *classification* construite au préalable (« segmentation » dans le langage du marketing). La segmentation regroupe les clients selon leurs similitudes ; la relation avec le client sera différente selon la classe (« segment ») dans laquelle il aura été rangé.

Cette démarche (classification, classement) constitue le contenu scientifique du marketing, discipline à base statistique. Pour servir les utilisateurs du système d'information l'entreprise doit segmenter la population qu'ils constituent et pratiquer un « marketing interne » qui a pour but non d'adapter les utilisateurs au système d'information, mais de définir ce que le système d'information doit faire pour répondre à leurs besoins.

Cependant l'organisation de l'entreprise repose souvent sur un postulat : la direction générale, où travaillent des personnes dont l'expérience s'est formée sur le terrain et qui ont été sélectionnées parmi les meilleurs agents du terrain, connaît les utilisateurs et sait ce qui est bon pour eux ; son infor-

2. Le besoin de communiquer à distance est aussi ancien que l'humanité mais il n'a pu s'exprimer sous forme de demande que lorsque le consommateur a eu connaissance de l'offre du service téléphonique.

3. Cette phrase surprendra ceux qui voient dans la demande le point de départ du raisonnement économique. Il suffit pourtant de s'examiner soi-même, par exemple lorsqu'on fait construire une maison, pour voir que la première formulation de la demande est maladroite et que le dialogue avec un architecte est nécessaire à l'expression du besoin.

À propos du mot « marketing »

Le mot « marketing » est entouré, dans le langage courant, de connotations négatives : on l'associe à l'utilisation indiscreète de la publicité, au « pied dans la porte » du vendeur insistant, au gavage des consommateurs avec des produits inutiles. C'est la faute des « gens du marketing », disait ainsi un participant au débat télévisé *C dans l'air*, si le laboratoire Merck n'a pas publié les résultats qui montraient que le Vioxx peut avoir des effets négatifs.

Nous prenons ici le mot « marketing » dans un sens technique qui désigne l'étude, puis la connaissance, des besoins des clients. Sous cette acception-là, les connotations négatives disparaissent puisqu'il s'agit d'aider à définir des produits utiles au consommateur.

mation est tenue à jour *via* la voie hiérarchique des responsables, directeurs régionaux ou directeurs d'agence, et par des contacts directs avec le terrain à l'occasion de missions d'inspection. Par ailleurs la direction générale, par ses réflexions, anticipe les évolutions du métier et possède une vue prospective que les utilisateurs ne peuvent pas avoir. Enfin, comme elle négocie avec les puissances externes, elle peut tenir compte de contraintes que les utilisateurs ignorent.

Le marketing interne n'aurait alors besoin ni d'observation des pratiques, ni de remontée d'alertes : les pratiques sont connues et les alertes ne feraient que manifester le mauvais esprit de personnes qui ignorent les contraintes auxquelles l'entreprise est soumise.

Mais ce postulat est erroné. Sauf exception les personnes nommées à la direction générale, quelles que soient leur qualité et la richesse de leur expérience, perdront en quelques mois une part de leur sensibilité opérationnelle à mesure qu'elles acquerront la sensibilité tactique qui est nécessaire à la direction générale. Par ailleurs la voie hiérarchique ne s'exprime pas toujours avec la précision et la vigueur des personnes du terrain car les dirigeants locaux ménagent leur image auprès de la direction générale. Les missions d'inspection, instructives sans doute, ont toujours un caractère artificiel : personne ne dit le fond de sa pensée devant un inspecteur.

Les responsables de la direction générale chargés de concevoir les évolutions du métier et de l'organisation doivent donc admettre la nécessité du marketing interne, mais cela leur est difficile car ils craignent qu'il ne compromette la légitimité de leurs décisions.

* *

Il est difficile de faire admettre l'idée du « marketing du système d'information », interne à l'entreprise, orienté vers les utilisateurs et impliquant que l'on segmente leur population. Il est clair pourtant que l'utilisation du système d'information n'est pas la même selon la fonction de l'utilisateur : lorsqu'on parle de « l'utilisateur » au singulier on néglige cette diversité.

Le refus de savoir

Un MOAD avait fait réaliser une enquête de satisfaction auprès des utilisateurs du système d'information. Les résultats furent présentés au CSSI. Ils faisaient apparaître des défauts dans le fonctionnement du système d'information, en particulier dans l'impression des courriers.

Le directeur responsable émit alors le petit bijou de rhétorique que voici (citation textuelle) : « Nous connaissons déjà ce problème et nous avons pris les mesures nécessaires, mais les gens du terrain n'y ont rien compris. Il faut dire qu'il y a du mauvais esprit... »

Le directeur général en conclut que celui qui avait présenté les résultats avait « exagéré » : c'est ainsi que l'on décourage la remontée directe d'informations du terrain vers la direction générale.

« Segmenter », ce n'est pas « personnaliser ». C'est d'abord définir dans la population des classes d'équivalence qui regroupent les utilisateurs aux besoins analogues (« segments »), puis identifier les paramètres selon lesquels on peut classer un utilisateur particulier, enfin définir les actions à entreprendre envers chaque segment.

On considère alors l'utilisateur non pas en tant que « personne » dans son individualité ineffable, mais seulement en tant qu'il appartient à une sous-population que l'on a au préalable définie. Il en est de même, notons-le, lorsque l'entreprise segmente la population de ses clients : la « personnalisation » de la relation avec ceux-ci résultera d'une segmentation, et non d'une adéquation à leur personnalité individuelle. Il en est de même aussi, quoique l'on puisse en dire, de la relation entre médecin et patient : quel que soit le contenu humain éventuel de cette relation, le diagnostic n'est rien d'autre que le classement du patient dans une nomenclature de pathologies, classement d'où résultera ensuite la prescription.

Il convient de percevoir les apports et les limites de la segmentation, notamment l'écart qui la sépare de la personnalisation. Certaines personnes en effet refusent la segmentation parce qu'elles préféreraient la personnalisation ; elles voient que la segmentation n'est pas une personnalisation véritable, mais elles ne conçoivent pas que celle-ci serait impossible et que la segmentation apporte un « plus » indispensable.

La segmentation des utilisateurs du système d'information doit se fonder sur une classification de leurs activités et compétences ; elle doit pouvoir évoluer selon les changements de l'organisation et des outils offerts par le système d'information. Elle doit être complétée par la remontée d'alertes fournie par exemple par une enquête périodique de satisfaction analogue à celle que l'entreprise fait auprès de ses clients.

Chaque entreprise possédant sa propre organisation, on ne peut pas proposer de segmentation ayant *a priori* une portée générale. Nous proposerons ici des critères qui indiquent comment, selon nous, les segments peuvent être définis.

Nous parlerons d'abord des besoins propres à l'ensemble des utilisateurs ; puis nous distinguerons les utilisateurs selon leur fonction dans l'entreprise, leur position par rapport aux processus de production et par rapport au système d'information.

12.1.1 Besoins généraux

L'utilisateur a d'abord besoin que son accès au système d'information *fonctionne* : il faut donc que la plate-forme technique lui procure une performance convenable en termes de fiabilité, de délai d'affichage, de délai de remise en route en cas de panne etc. Il faut que les divers matériels utilisés localement (poste de travail, serveur, modem, imprimantes, scanners, téléphone et PABX etc.) fonctionnent correctement et communiquent entre eux sans difficulté.

L'utilisateur a besoin de savoir à qui s'adresser en cas de problème, soit pour un dépannage en cas de mauvais fonctionnement, soit pour une aide lorsqu'il rencontre la limite de ses propres compétences. L'aide doit pouvoir lui être fournie dans un délai assez court pour ne pas compromettre l'exécution du travail, donc soit par une personne géographiquement proche (« support de proximité »), soit par des experts travaillant en réseau (*help desk* et réseau d'experts) et disposant des outils nécessaires (prise de main à distance, téléchargement, télédistribution).

Les utilisateurs doivent recevoir une formation suffisante pour pouvoir utiliser les outils mis à leur disposition : il serait dérisoire de fournir Excel à quelqu'un qui n'a jamais été formé au tableur, ou de donner un accès à la messagerie à quelqu'un qui ne saurait pas ouvrir une pièce jointe. Lorsque de « grandes applications » sont déployées, une formation doit être dispensée dans les jours qui précèdent l'installation et par la suite il faudra des « piquûres de rappel ». Des outils de formation personnelle doivent être fournis aux utilisateurs soit en réseau, soit sur CD-Rom, et comporter des exercices permettant à chacun de vérifier son niveau de connaissance.

Ces recommandations doivent s'appliquer à toutes les personnes de l'entreprise et même aux cadres de haut niveau. Il ne doit pas être admis qu'un dirigeant ne mette jamais les doigts sur un clavier car il s'isolerait de l'entreprise : celui qui n'est pas *in* se mettant *out*, tout le monde doit accéder au système d'information.

Lors de l'accès au système d'information, l'utilisateur doit être identifié, son identité doit être authentifiée par un mot de passe, il doit bénéficier ensuite des habilitations correspondant à sa fonction. Ces procédures doivent être administrées de façon à garantir la sécurité du système d'information sans gêner les personnes.

L'informatique de communication doit être mise à la disposition de chacun : chaque personne doit avoir une boîte aux lettres dans la messagerie électronique⁴, un annuaire électronique doit permettre de retrouver les per-

4. Il est possible d'ouvrir des forums, mais il faut désigner un animateur si l'on veut éviter qu'ils ne se dégradent.

sonnes et les entités de l'organisation ainsi que leurs coordonnées ; chaque personne doit disposer d'un agenda électronique consultable en réseau et facile à répliquer sur un Palmtop ; la documentation professionnelle de l'entreprise doit être placée sur un Intranet tirant parti des ressources de l'hypertexte ; des aides contextuelles doivent être accessibles lors de l'utilisation des applications ; des outils de rédaction coopérative doivent être fournis pour préparer et valider les documents de référence ; des outils de diffusion sélective doivent fournir aux salariés les informations qui leur sont nécessaires (extraits de revues de presse et d'articles de la presse professionnelle, commentaires de lecture etc.).

Les divers processus administratifs (préparation du budget, demandes de mutation, de congé, d'équipement, de réparation etc.) doivent être équipés de workflows qui permettent le contrôle du délai du traitement des affaires et la traçabilité des dossiers.

12.1.2 Besoins propres à la fonction dans l'entreprise

On distingue dans une entreprise trois grandes fonctions : l'*exécution* ; l'*organisation* (on peut dire aussi contrôle ou animation) de l'exécution ; la *conception* des tâches à exécuter.

Cette classification n'a aucune relation avec le niveau social des personnes : dans une compagnie aérienne un commandant de bord est un exécutant, pourtant il se peut qu'il soit mieux payé que le président lui-même⁵ ; il arrive qu'un chirurgien, qui est lui aussi un exécutant, ait plus de prestige social que le directeur de l'hôpital.

* *

L'*exécution* (conseiller dans une agence de l'ANPE, commandant de bord et hôte de l'air dans une compagnie aérienne, agent du centre d'appel, *trader* dans une banque, acheteur d'une chaîne de grands magasins, commerciaux, administratifs chargés de la gestion des ressources humaines, de la comptabilité etc.) occupe de l'ordre de 90 % des effectifs de l'entreprise. Il s'agit des personnes qui font tourner le processus de production, sont en relation directe avec les clients et les fournisseurs ou manipulent la matière première et les produits intermédiaires pour les transformer.

Le système d'information doit leur fournir à chaque instant les données nécessaires à l'action en cours, les espaces où elles peuvent saisir les données qu'il faut entrer dans le système, les commandes qui leur permettent de lancer des traitements. Le contrôle d'accès (identification, authentification) doit être rigoureux mais aussi simple que possible (*single sign on*) ; les habilitations doivent être compatibles avec le travail à réaliser ; il ne doit pas exister de double saisie et le système d'information doit assurer lui-même la navigation entre les applications dont la personne a besoin ; la performance informatique doit être convenable (pas de pannes, affichage rapide).

5. À Air France, du temps de la présidence de Christian Blanc, le président percevait le huit-centième salaire de la compagnie.

On peut distinguer plusieurs catégories parmi les travaux d'exécution : activités au contact des clients et des fournisseurs ; activités de production internes à l'entreprise ; activités administratives, dites « de support » ; activités d'assistance ; activités externes.

Contact avec les clients et fournisseurs

Ce sont les activités des commerciaux, acheteurs, opérateurs des centres d'appel, conseillers de l'ANPE, du personnel des agences de voyage, des agences bancaires et des bureaux de poste, des hôtesse et stewards du transport aérien etc.

Elles impliquent une part de négociation : il faut que l'agent puisse comprendre ce que lui dit quelqu'un qui ne parle pas le langage de l'entreprise. Le système d'information doit fournir la segmentation des clients et une aide au diagnostic (exemple : *life time value*⁶ du client dans le transport aérien) ; il doit garder mémoire des relations avec les clients et fournisseurs (contrat client et compte client, contrat d'achat et compte fournisseur, relevé des éventuels incidents).

Lorsque le travail se fait au guichet ou au centre d'appel le système d'information doit permettre d'identifier le client et de parcourir les rubriques du dossier selon l'ordre de la conversation. Il doit permettre aussi la prise de commande immédiate, ce qui implique la vérification du compte client et de la disponibilité du stock ainsi que la rédaction et le lancement de la commande.

Le système d'information peut présenter deux types d'interface : l'une professionnelle (et austère) pour le traitement rapide des cas banals, l'autre plus ergonomique et offrant une aide en ligne pour traiter les cas compliqués (transport de civière ou location de chambre d'hôtel dans le transport aérien). Les commerciaux qui négocient des contrats chez les clients sont dotés d'un ordinateur portable et équipés pour la connexion sans fil au système.

Production interne à l'entreprise

Ces activités sont celles du *back office*, des ateliers de fabrication, de gestion des stocks, manutention, maintenance, programmation informatique, celles des pilotes dans le transport aérien, des services administratifs d'une agence locale de l'ANPE etc. Il s'agit d'une production au sens strict, qui fournit aux processus de l'entreprise les « livrables » intermédiaires qui les font progresser.

Le système d'information balise les processus de production de l'entreprise en automatisant le transfert de messages d'une étape à l'autre, en spécifiant les opérations propres à chaque étape (*use cases*), en assurant la traçabilité des dossiers en cours. Il aide à établir et gérer le calendrier des opérations de maintenance en relation avec les contraintes de la production, et fournit aux techniciens des fiches de travail⁷.

6. Valeur anticipée de la consommation du client. La « *life time value* » d'un étudiant qui voyage en classe économique peut être importante, car dans quelques années il sera cadre, alors que celle d'un président-directeur général en fin de carrière qui voyage en première classe est faible, car dans quelques années il sera à la retraite et volera en classe économique.

7. Les techniciens de maintenance de l'aéronautique sont équipés d'un lecteur de CD-

Les courriers sont préformatés, le système d'information assurant la gestion des listes ainsi que l'impression et l'envoi des diffusions de masse, puis la gestion des rappels, relances et mises en demeure.

Le système d'information doit fournir automatiquement les informations sur l'avancement du processus et émettre des alarmes en cas de retard, de goulet d'étranglement ou de sous-charge de travail.

La gestion des stocks et des achats nécessite un système spécifique lié d'une part aux processus de production et de maintenance, d'autre part à la comptabilité.

Logistique

Ce terme recouvre la gestion de ressources humaines, la comptabilité, la finance, l'audit, l'administration des processus etc. : ces activités amènent aux unités de production des ressources humaines, matérielles, financières, de maîtrise de la qualité.

L'administrateur d'un processus opérationnel, tout comme l'animateur d'un outil d'informatique communicante (messagerie, forums), doit recevoir quotidiennement des indicateurs de volume, de qualité, de délai ; il doit pouvoir émettre des alarmes sur le fonctionnement du processus et intervenir (par messagerie, par téléphone) pour corriger les mauvaises pratiques.

Les personnes responsables de la gestion des ressources disposent d'indicateurs qui permettent de vérifier l'adéquation des ressources aux besoins. L'analyse comparée des compétences disponibles et des compétences nécessaires alimente les opérations de recrutement et de formation.

Assistance

Ce terme recouvre le « help desk », l'expertise et le dépannage : *a priori* les effectifs sont modestes, à l'exception du support bureautique de proximité qui dans une entreprise de service occupera une personne pour 50 à 100 utilisateurs.

L'assistance répond aux questions et demandes des utilisateurs portant sur la mise en œuvre des outils professionnels, le dépannage, l'expertise. Elle comporte un premier niveau local (assistance bureautique, encadrement) ; puis un centre d'appel (« help desk ») possédant les compétences pour résoudre certains problèmes (dépannage, questions courantes) ; enfin un réseau d'experts spécialisés capables de traiter les questions les plus difficiles (expertises technique, commerciale, juridique). Elle gère des contrats avec les fournisseurs pour les opérations de dépannage.

La mise en réseau des activités d'assistance nécessite des centres d'appel équipés en matériel spécialisé, des systèmes documentaires, des tables d'adressage permettant de trouver les ressources d'expertise. La mémorisation des incidents nourrit la réflexion sur l'organisation de l'entreprise et la formation des personnes.

Utilisateurs externes

Les activités externes (c'est-à-dire exécutées par des agents extérieurs

Rom et de lunettes sur lesquelles s'affichent le dessin des équipements et la liste des opérations à effectuer.

à l'entreprise) pénètrent de plus en plus le système d'information. L'évolution de l'*e-business* a fait du client un utilisateur du système d'information, dans le cadre de ses habilitations. Il en est de même des partenaires et des fournisseurs.

Le *client* a besoin que le service lui soit fourni « sans couture », que l'entreprise ait articulé les divers canaux par lesquels elle communique avec lui (face-à-face, téléphone, messagerie, courrier, Web etc.) de telle sorte que le client n'ait pas à donner plusieurs fois les mêmes informations et qu'il ait donc toujours devant lui quelqu'un qui soit au courant de l'état de ses relations avec l'entreprise. Il doit pouvoir trouver sur la Toile une interface avec l'entreprise rédigée dans sa langue naturelle et professionnelle ; y trouver les informations relatives à sa relation avec l'entreprise (traçabilité des affaires) ainsi qu'une assistance à l'utilisation des produits de celle-ci ; de la documentation technique, des masques de saisie permettant une prise de commande facile. Il doit pouvoir avoir au téléphone des personnes au courant de ses affaires avec l'entreprise. Il doit éventuellement pouvoir bénéficier d'une segmentation tarifaire.

L'entreprise doit pouvoir fournir à ses grands clients une facture sur support approprié ainsi que des services annexes (tenue de la comptabilité voyage par un transporteur aérien ; facture sur CD-Rom au format comptable de l'entreprise etc.).

Pour le *partenaire*, les systèmes d'information doivent pouvoir interopérer, le transcodage et le traitement des anomalies étant exécutés soit automatiquement, soit par des équipes spécialisées. La relation doit être assez transparente pour garantir l'honnêteté du partage des dépenses et des recettes. L'organisation du partenariat doit avoir été préparée par une *ingénierie d'affaire* qui précise les responsabilités de chacun et organise l'interopérabilité des systèmes d'information.

Les relations avec les *fournisseurs* sont dans certaines industries (automobile, aéronautique) automatisées en utilisant l'EDI (Échange de données informatisé) : le fournisseur, informé du niveau des stocks de matières premières ou produits semi-finis dans l'entreprise, les réapprovisionne automatiquement.

* *

Le *contrôle de l'exécution*, ou encore son organisation et son animation (directeur d'agence dans une banque, chef d'escale dans une compagnie aérienne, directeur régional ou chef de division à l'INSEE, proviseur dans un lycée etc.) occupe environ 10 % des effectifs.

Il est assuré sur le terrain par le personnel d'encadrement qui vérifie la qualité et la quantité des travaux et prend les mesures nécessaires pour que l'exécution se déroule efficacement.

Le *manager opérationnel* doit disposer d'indicateurs quotidiens permettant de contrôler la qualité du travail et la bonne affectation des ressources : indicateurs de volume, de délai, de temps de travail, indicateurs composites (productivité, degré de réalisation d'un programme etc.), alertes. Le système d'information doit lui fournir ces indicateurs de façon automatique,

sans intervention humaine (génératrice de délais et parfois de manque d'objectivité). Il doit permettre de comparer la performance de l'unité avec celle des autres unités.

Ici la fraîcheur et l'ordre de grandeur comptent plus que la qualité : ces indicateurs sont établis à partir de données brutes, non redressées des erreurs ou biais qui entachent les données opérationnelles. Leur présentation doit faciliter leur consultation et leur interprétation.

* *

La *conception* (directeur général, directeurs régionaux, directeurs d'administration centrale entourés de leurs collaborateurs immédiats ; direction du marketing, directeur commercial, direction de la production etc.) occupe de l'ordre de 1 % des effectifs. Il s'agit de définir la stratégie de l'entreprise en termes de produits, commercialisation, distribution, partenariats, tarification et de mettre au point les dispositions de détail nécessaires à son application. Il faut distinguer plusieurs niveaux dans la conception des tâches : les dirigeants stratégiques (comité de direction), les dirigeants régionaux ou locaux et responsables de lignes de produits, les chargés d'étude, experts, chercheurs et planificateurs.

Comité de direction

Les dirigeants (président-directeur général, directeur général, DGA et directeurs centraux) doivent disposer de données mensuelles corrigées des biais et erreurs statistiques, de tableaux de bord sélectifs, commentés et présentant sous forme graphique des séries chronologiques corrigées des variations saisonnières qui décrivent l'essentiel de l'activité de l'entreprise : part de marché et chiffre d'affaires dans les divers segments de clientèle et les divers produits, avec l'explication des incidents exceptionnels ; situation financière et estimation du résultat de l'entreprise ; indicateurs de productivité et de compétitivité etc. (voir page 379).

Ces tableaux de bord doivent permettre aux dirigeants de ne pas avoir à s'interroger ni se disputer sur les sources statistiques. Ils doivent pouvoir connaître la fonction de coût de l'entreprise et être complétés par des rapports d'études (veille concurrentielle, veille technologique).

Responsables

Les directeurs régionaux ou locaux, les chefs de lignes de produit disposent d'outils d'étude sous forme de *datawarehouses* (données relatives à la production et au marché selon le croisement de plusieurs nomenclatures prédéfinies, tableaux préformatés, voir page 299) ; une équipe d'experts nationale traite les requêtes complexes.

Chargés d'étude, experts et planificateurs

Les chargés d'étude de la direction du marketing disposent d'un *datawarehouse* qui leur permet d'étudier la segmentation de la clientèle, la concurrence, les parts de marché, les tendances économiques ; ils exploitent les enquêtes de satisfaction auprès des clients, les comptes rendus du service après vente et de l'assistance à la force de vente etc. Ils utilisent en outre des outils de *datamining* (voir page 296) pour identifier la cause des incidents et anomalies. Des planificateurs préparent le programme des vols à Air France,

dimensionnent le réseau à France Telecom, les plans d'équipement dans une entreprise industrielle, la liste des projets informatiques ; ils ont besoin de statistiques pour asseoir leur décision (marché potentiel, matrice de trafic attendu etc.) et de workflows pour assurer le partage des informations entre parties concernées et la convergence du processus de décision.

12.1.3 Une segmentation des données

Les utilisateurs sont dans des positions diverses par rapport aux données :

- certains (surtout ceux qui sont au contact des clients et fournisseurs, mais pas eux seuls) saisissent des données relatives aux événements du monde de la nature, et l'exactitude de la saisie doit être vérifiée ;
- d'autres (surtout ceux qui travaillent dans la production) produisent des documents internes qui préparent la réponse aux événements externes (factures, bulletin de paie etc.) ;
- enfin, certaines données restent internes (données de pilotage, données statistiques) : elles sont destinées à alimenter le contrôle des processus ou la réflexion stratégique.

Les exigences en matière de qualité sont en effet différentes selon les trois cas.

Données externes

L'information provient de l'extérieur du système d'information. Elle est d'origine soit externe à l'entreprise (commande d'un client), soit interne à l'entreprise (nouveau produit mis au catalogue, exécution d'une commande).

Lorsque cette information est introduite dans le système il faut vérifier son exactitude et sa vraisemblance (messages d'erreur, messages d'anomalie) : quelle que soit la qualité du système d'information, il ne pourra donner que des résultats faux si les données externes sont fausses : *garbage in, garbage out*.

Données générées

Ce sont des données que le système fabrique à partir des données externes et qu'il doit elles aussi mémoriser : salaire payé à un employé, montant d'un stock, montant d'une facture etc. À toute donnée générée est associée une formule de calcul qui implique des conditions de génération et des contraintes de cohérence.

Données statistiques

Ce sont des données que le système construit à partir des données externes ou des données générées et qu'il n'est pas toujours nécessaire de mémoriser, car on pourra les calculer de nouveau si besoin est : nombre de commandes reçues chaque jour, chiffre d'affaire cumulé depuis le début de l'année avec un client, listes de personnes à convoquer etc. À toute donnée statistique est associée une formule d'élaboration. Certaines de ces formules supposent de redresser la donnée pour la corriger des biais de collecte ou des conventions comptables.

12.2 Organisation et responsabilités

Pour comprendre comment s'organisent la conception et l'exploitation d'un système d'information, il faut tirer au clair deux expressions du vocabulaire professionnel, « maîtrise d'ouvrage » et « maîtrise d'œuvre ».

Alors que dans le langage courant, les mots « œuvre » et « ouvrage » sont à peu près synonymes, ces deux expressions se réfèrent à leurs acceptions originelles mais anciennes et donc souvent oubliées.

Ouvrage : Définition, financement, organisation de l'action et du travail en vue de la réalisation d'un produit.

Œuvre : Produit fourni par un travail et propre à un usage.

La façon dont le travail est mené (ouvrage) a une influence sur l'œuvre (délais, qualité, satisfaction des clients etc.).

En bref : *ouvrage* désigne le *processus* de production, *œuvre* désigne le *produit*. Le maître d'ouvrage est responsable d'un processus de production ; le maître d'œuvre est responsable d'un produit.

Chaque entreprise, et dans l'entreprise chaque direction et chaque service, sont donc à la fois maître d'ouvrage de leur organisation, de leurs processus, et maître d'œuvre de leur produit. Considérer une direction comme maître d'ouvrage, c'est considérer la façon dont elle fonctionne, dont elle s'organise pour produire, c'est orienter le regard vers l'intérieur de cette direction. La considérer comme maître d'œuvre, c'est orienter le regard vers ses produits, vers ce qui sort de cette direction.

Que chaque entité ait à la fois un dehors et un dedans, c'est une banalité. Pourtant il n'est pas facile d'assimiler les définitions ci-dessus et elles méritent qu'on leur consacre attention et réflexion. Nous allons déployer leurs implications.

Maître d'ouvrage : le maître d'ouvrage (MOA) est dans le monde de l'entreprise une *personne morale* (entreprise, direction etc.), une entité de l'organisation. Ce n'est jamais une personne physique même si, dans le langage courant qui est quelque peu relâché (mais il serait pédant de refuser ce relâchement) on dit souvent « Untel est maître d'ouvrage ». Qu'il soit MOAD, MOAO et AMO (ces acronymes seront définis ci-dessous) en fait il *représente* une maîtrise d'ouvrage.

La distinction entre personne morale et personne physique est importante au plan juridique : c'est l'entreprise, la direction etc. qui passent un contrat avec un fournisseur ; ce contrat est signé par une personne physique mais elle ne fait que représenter la personne morale : le contrat reste valable même si le directeur qui l'a signé est remplacé par une autre personne.

Maître d'œuvre : le maître d'œuvre (MOE⁸) est lui aussi une *personne morale* (entreprise, direction etc.) garante de la qualité du produit qu'elle livre à son client. Il a lors de la conception du produit un devoir de conseil vis-à-vis du MOA, car celui-ci doit pouvoir tirer le meilleur parti de l'état de l'art.

8. L'acronyme MOA se lit indifféremment « maître d'ouvrage » ou « maîtrise d'ouvrage » ; de même, MOE se lit « maître d'œuvre » ou « maîtrise d'œuvre ».

Lorsque le produit est compliqué il peut être nécessaire de faire appel à plusieurs fournisseurs. Le MOE assure leur coordination ; il veille à la cohérence des fournitures et à leur compatibilité. Il coordonne l'action des fournisseurs en contrôlant la qualité de leur produit, en assurant le respect des délais fixés par le MOA et en minimisant les risques. Il est responsable de la qualité technique de la solution. Il doit, avant toute livraison au MOA, procéder aux vérifications nécessaires (« recette usine »).

À retenir : Maîtrise d'ouvrage (MOA) et Maîtrise d'œuvre (MOE) sont des *personnes morales*, des entités de l'organisation. La MOA est client de la MOE à qui elle passe commande d'un produit nécessaire à son activité. La MOE fournit ce produit. Soit elle le réalise elle-même, soit elle passe commande à un ou plusieurs fournisseurs qui élaborent le produit sous sa direction et sa responsabilité.

La relation entre la MOA et la MOE est définie par un contrat (souvent implicite, mais néanmoins réel) qui précise leurs engagements mutuels (Peaucelle [157]).

* *

Contrairement à une opinion étonnamment répandue, la distinction entre MOA et MOE n'est pas « franco-française ». Seul le vocabulaire est propre à notre pays. Aux États-Unis, les métiers d'une entreprise comportent évidemment des experts de la modélisation du système d'information : on les appelle *IT analysts, business technologists, business analysts*⁹ (*organisers*) en Grande-Bretagne, *IT coordinators* au Japon, et certaines universités forment à cette spécialité. Leurs fonctions sont celles des « maîtres d'ouvrage délégués » et « maîtres d'ouvrage opérationnels » que nous décrirons ci-dessous.

Comme toute entité de l'organisation est à la fois maître d'ouvrage (de son processus de production) et maître d'œuvre (de son produit), il est parfaitement correct de dire que la DSI est maître d'ouvrage de son propre processus de production. Mais dans le cas particulier du système d'information le partage des rôles est clair : la DSI sera maître d'œuvre (du produit que constitue le système d'information) et les autres métiers de l'entreprise sont chacun maître d'ouvrage (de ses processus de production, que le système d'information doit outiller)¹⁰.

12.2.1 Fonctions dans la maîtrise d'ouvrage

Nota Bene : Le vocabulaire de la maîtrise d'ouvrage n'est pas encore stabilisé, certaines entreprises étant parfois déconcertées devant cette spécialité

9. Le titre de CIO (*Chief Information Officer*) ayant souvent été attribuée au DSI, la personne qui anime l'expertise de l'agence en maîtrise d'ouvrage est nommée CTO (*Chief Technology Officer*), l'acronyme *IT* étant ainsi réparti entre ces deux fonctions de façon quelque peu paradoxale.

10. Cette formulation correspond au cas où le système d'information est conçu et exploité par la DSI de l'entreprise ; il faut transposer le vocabulaire si celle-ci a confié cette responsabilité à une autre entreprise (*outsourcing*).

qui est nouvelle pour elles. Nous utiliserons dans cet ouvrage le vocabulaire des entreprises où la réflexion sur la maîtrise d'ouvrage nous semble la plus avancée.

On distingue dans la MOA six fonctions : le maître d'ouvrage stratégique (MOAS) ; le maître d'ouvrage délégué (MOAD) ; le maître d'ouvrage opérationnel (MOAO) ; l'assistant à maîtrise d'ouvrage (AMO) ; l'expert métier ; enfin l'utilisateur, au service duquel se trouvent les cinq autres fonctions.

Le maître d'ouvrage stratégique (MOAS)

À la tête de toute entité se trouve un dirigeant : président-directeur général ou directeur général pour l'entreprise, DGA ou directeur pour une direction de l'entreprise, chef de service pour les unités qui constituent la direction etc. Les appellations varient d'une entreprise à l'autre, mais on comprend que nous pensons ici à l'organisation hiérarchique.

Le « maître d'ouvrage stratégique » (MOAS) est une personne physique ; c'est le dirigeant de l'entité, le patron, ou encore le « stratège ». C'est lui qui prend en définitive les décisions importantes concernant l'entité, qui arbitre les différends entre ses collaborateurs, qui signe le contrat avec la MOE. Comme le système d'information est, pour la plupart des entités, à la fois concrétisation de la stratégie et condition de sa mise en œuvre, des décisions essentielles seront prises par le MOAS notamment lors du lancement des grands projets. Cependant le MOAS *n'est pas*, en général, un expert en matière de système d'information. Même s'il a été un expert dans une vie antérieure, ses responsabilités présentes lui interdisent de se tenir au courant de l'état de l'art. C'est pourquoi il se fait assister par un maître d'ouvrage délégué (MOAD) auquel il *délègue* (c'est le sens même de l'expression) le soin, le souci de l'expertise dans le domaine du système d'information.

Le principe qui sous-tend cette organisation est la *séparation de l'expertise et de la décision* (voir page 421). L'expert n'a pas à porter le poids de la décision, mais il la prépare en fournissant au décideur les éléments d'information nécessaires ; le décideur n'a pas à se tenir au courant de l'état de l'art, mais il est attentif aux priorités de l'entreprise, vigilant envers son environnement, et il peut ainsi arbitrer entre les solutions possibles en situant dans un contexte plus large les informations que lui apporte l'expert.

Le maître d'ouvrage délégué (MOAD)

Le « maître d'ouvrage délégué » (MOAD) est une personne physique soit seule, soit à la tête d'une équipe si l'entité est importante. Il assiste le MOAS dont il est un des proches collaborateurs.

Sa fonction est de veiller à la qualité du système d'information de l'entité, tant dans sa conception que dans la façon dont il est utilisé. Il assiste le MOAS en lui fournissant les éléments nécessaires à la décision et en veillant à ce que l'on appelle l'« alignement stratégique » du système d'information.

Le MOAD est, au sein de l'entité, responsable de la qualité des méthodes utilisées pour l'expression des besoins, l'explicitation des priorités, la sélection des projets, la réalisation des études préalables, la conduite de projet, la recette finale, enfin il doit animer le bon usage du système d'information existant. Ses interlocuteurs naturels au sein de l'entité sont les chefs de ser-

vice et les MOAO (voir ci-dessous). Son interlocuteur naturel au sein de la DSI est le responsable de domaine (figure 12.1).

Le MOAD assure une « veille système d'information » : il se tient informé de ce que font les entités comparables à la sienne pour s'assurer que celle-ci ne prend pas de retard par rapport à l'état de l'art, ou pour formuler des suggestions qui lui permettront de prendre de l'avance.

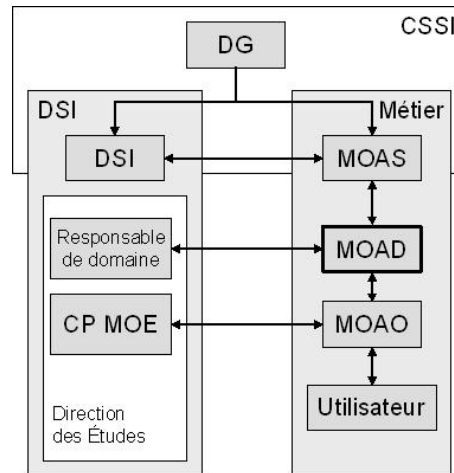


FIG. 12.1 – Place du MOAD

C'est un organisateur plus qu'un informaticien (en Grande-Bretagne, on l'appelle justement *organiser*), mais il est bon qu'il ait assez de connaissances en informatique pour être face à la DSI un *client compétent*, qui sache se faire comprendre et comprenne ce qu'on lui dit.

Souvent, le MOAD sera un ancien informaticien : l'expérience montre qu'un informaticien, s'il n'a pas été déformé par une spécialisation trop étroite, peut assimiler en quelques mois les caractéristiques essentielles du métier pour lequel il travaille alors qu'il serait plus difficile à un expert du métier de s'acclimater à l'informatique.

Le maître d'ouvrage opérationnel (MOAO)

Le « maître d'ouvrage opérationnel » (MOAO) est, dans l'entité, un expert qui connaît à fond l'un des grands processus du métier ou, si le système d'information est organisé en applications, qui connaît parfaitement une de ces applications.

Il sait, quand on veut faire faire à l'application des choses nouvelles, s'il sera possible d'y parvenir en recourant à un paramétrage ou s'il faudra un développement important ; il connaît les référentiels utilisés, les règles de gestion, les traitements en batch, les échanges avec les autres applications etc. Il recueille les demandes des utilisateurs et établit ou supervise les spécifications générales. Il a pour interlocuteur naturel à la DSI un « chef de projet MOE » avec lequel il entretient une relation assidue.

Le MOAO possède une expertise précieuse. Souvent, c'est un homme seul au caractère solide mais un peu bourru, un ours solitaire (voir page 197) qui n'aime pas à partager ses connaissances. Il peut en résulter une catastrophe s'il tombe malade ou quand il prend sa retraite. Le MOAD doit veiller à la bonne utilisation des méthodes par les MOAO, ainsi qu'à la qualité de leurs relations avec le chef de projet MOE. Il doit aussi inciter le MOAO à partager son savoir.

L'assistant à maîtrise d'ouvrage (AMO)

Le travail que doivent réaliser un MOAD ou un MOAO comporte des périodes de pointe lorsqu'il faut spécifier, suivre ou recetter un gros projet ; lorsqu'il faut concevoir ou mettre en place des méthodes nouvelles ; lorsqu'il faut mettre en œuvre une expertise que l'entité ne possède pas. Alors l'entité fait appel, pour réaliser les travaux dévolus à la maîtrise d'ouvrage, à des consultants ou à des personnes que la DSI mettra à sa disposition : elles rédigeront les spécifications, tiendront les tableaux de bord, rédigeront les comptes rendus, pointeront l'exécution des décisions, assisteront le directeur de projet etc. On appelle ces personnes « assistants à maîtrise d'ouvrage » (AMO).

L'expert métier

Pour garantir l'adéquation du système d'information à la pratique professionnelle, il faut recueillir une information sur les conditions du travail quotidien. Le MOAO s'adjoit à cette fin des « experts métier » venus du terrain et qui contribuent au travail de spécification et de recette.

Les indications fournies par ces experts métier sont indispensables mais il faut savoir les interpréter : des praticiens ne sont pas les mieux placés pour définir les priorités ni pour distinguer l'essentiel du secondaire.

L'utilisateur

La finalité du système d'information, c'est d'être un outil efficace entre les mains de l'utilisateur final. Cependant la population des utilisateurs est souvent trop nombreuse, trop dispersée et trop accaparée par le travail quotidien pour participer à la conception du système d'information. Elle sera représentée dans cette tâche par les experts métier.

Les utilisateurs doivent recevoir une formation peu de temps avant que le nouveau produit ne soit mis en service, et cette formation devra être renouvelée par des « piqûres de rappel » périodiques ; leurs avis doivent être recueillis lors du déploiement du produit et aussi lors de son exploitation courante par des enquêtes périodiques.

La coordination des maîtrises d'ouvrage

La maîtrise d'ouvrage, telle que nous l'avons définie, relève de chacun des métiers de l'entreprise : sont maîtres d'ouvrage du système d'information la direction commerciale, la direction de la production, la direction technique, la direction de la recherche etc¹¹.

Cependant le système d'information concerne non chaque métier séparément, mais l'ensemble de l'entreprise. Sa cohérence exige que les divers

11. La DSI, elle, est maître d'ouvrage de la plate-forme technique.

métiers utilisent un même référentiel pour l'identification des personnes, pour les produits et clients ainsi que pour l'organisation. Par ailleurs certains processus passent d'un métier à l'autre (la commande d'un client passe par le commercial, puis par la production, la comptabilité, la livraison etc.).

Les diverses maîtrises d'ouvrage utilisent des méthodes analogues pour préparer les projets et rédiger les spécifications. Le poste de travail, l'informatique de communication sont partagés par tous. Enfin, la gestion du patrimoine de l'entreprise suppose un arbitrage ultime pour sélectionner, parmi les projets que proposent les métiers, ceux qui seront effectivement réalisés. Cette sélection relève du directeur général qui est ainsi le MOAS suprême de l'entreprise. Il est assisté par une MOAD spécifique qui prépare ses arbitrages et sera chargée de la coordination et de l'animation des diverses MOA. Nous l'appellerons « Coordination des maîtrises d'ouvrage ¹² ». Elle assiste le directeur général par ses avis, ses expertises, et elle a le devoir de l'alerter lorsque des risques ou opportunités se présentent. Le DSI est son interlocuteur naturel.

Le Comité Stratégique du Système d'information (CSSI)

Les décisions essentielles concernant le système d'information sont prises lors de réunions du comité directeur de l'entreprise dans une formation spécifique que l'on nomme « Comité Stratégique du Système d'Information » (voir page 423).

12.2.2 Fonctions dans la maîtrise d'œuvre

La MOE s'identifie le plus souvent avec la DSI de l'entreprise (certaines entreprises externalisent leur informatique et alors la MOE est exercée par une entreprise comme EDS, IBM etc. : mais ce n'est pas le cas que nous considérons ici).

L'organisation interne d'une DSI est délicate, comme celle de toutes les entités dont la compétence est l'actif principal. Les savoirs qu'elle doit posséder sont multiples, les spécialités se diversifient, et le DSI doit répondre à deux questions des plus difficiles : « Quelles sont les compétences que nous devons cultiver en interne, quelles sont celles que nous devons nous procurer à l'extérieur? » ; « Comment faire pour former et conserver les compétences dont nous avons besoin en interne? »

Le DSI

Le responsable de la maîtrise d'œuvre, c'est le DSI de l'entreprise. Il est par ailleurs maître d'ouvrage stratégique de la plate-forme technique : sa mission est de fournir à l'entreprise un service informatique satisfaisant pour un bon rapport qualité/prix. Il est malencontreux que l'on ait si souvent donné au directeur informatique le titre de DSI (« directeur du ou des

12. Dans certaines entreprises, la coordination des maîtrises d'ouvrage est appelée « Maîtrise d'ouvrage stratégique ». Cette dénomination est correcte au plan logique (puisque cette unité assiste le directeur général qui est le « stratège » de l'entreprise), mais elle a l'inconvénient de laisser sans appellation le directeur d'un métier qui, lui, est le « stratège » d'une maîtrise d'ouvrage particulière. Nous conseillons donc de réserver à celui-ci l'expression « maître d'ouvrage stratégique ».

système(s) d'information »), puisque la responsabilité du système d'information est partagée entre les MOA (chargées de définir les fonctionnalités que le système d'information doit offrir) et la MOE (responsable de la réalisation et de l'exploitation). Il aurait mieux valu retenir « Directeur informatique », mais ici comme ailleurs nous nous plions à l'usage¹³.

La DSI comporte plusieurs directions (voir page 489 l'organigramme type d'une DSI) : direction des études, de la production, des télécoms, du soutien (*support*) aux utilisateurs etc. Elles concourent toutes à la production du service. Cependant, si l'on considère l'évolution du système d'information et non son exploitation quotidienne, c'est la direction des études qui a le rôle le plus important.

Une direction de l'architecture anime la veille technologique, veille à la cohérence de la plate-forme et prépare ses évolutions. Parfois elle fait partie de la direction des études mais il est préférable qu'elle soit rattachée directement au DSI car elle remplit auprès de lui les fonctions d'un MOAD.

Le DSI signe avec les MOAS des métiers les contrats relatifs à la qualité de service (« contrats de service ») et à la réalisation des projets (« fiches de synthèse de mission », voir page 463).

Le directeur des études

La direction des études est responsable de la maintenance du système d'information (« maintien en condition opérationnelle¹⁴ ») et de son évolution.

Le système d'information étant pour l'entreprise un *actif*¹⁵, son évolution résulte d'un investissement. Dans le cas particulier des logiciels cet investissement est nommé « développement », terme qui a été sottement jugé plus noble que « programmation » parce que « développeur » serait plus noble que « programmeur ».

Le responsable de domaine

À la direction des études, le responsable de domaine sera le correspondant privilégié d'un MOAD : il supervise l'ensemble des travaux concernant le système d'information d'un des métier de l'entreprise.

Le chef de projet MOE (CP MOE)

Le chef de projet maîtrise d'œuvre (CP MOE) assure la gestion de la maîtrise d'œuvre d'un projet particulier. C'est un chef de chantier, avec tout ce que cela implique de savoir-faire et d'énergie : il veille à la qualité des spécifications techniques, coordonne les fournisseurs, vérifie la qualité des fournitures, s'assure de la cohérence entre les diverses livraisons etc.

13. Certains DSI ont l'ambition d'assumer à eux seuls toute les responsabilités sur le système d'information, y compris celle de la maîtrise d'ouvrage. C'est faire prendre un risque à l'entreprise (voir page 572).

14. Les petits développements sont souvent appelés « maintenance évolutive », alors qu'il s'agit d'investissements et non de maintenance : mais comme il est souvent plus commode de les financer dans le cadre des contrats de « tierce maintenance applicative » (TMA) plutôt que de les considérer comme des projets on utilise souvent une dénomination trompeuse.

15. Même s'il n'est pas considéré comme un actif par la comptabilité, ici en retard sur l'économie.

12.2.3 Rôle de la maîtrise d'ouvrage

Il revient d'abord à la maîtrise d'ouvrage de définir les besoins du métier. Cette définition doit être adéquate aux besoins (pertinence), sobre, et respectueuse de la cohérence sémantique du système d'information.

Lorsque le système d'information existe, l'ensemble de ses fonctionnalités (processus, composants, interfaces homme-machine) constitue un patrimoine (le « portefeuille applicatif ») qui comme tout actif doit être *géré*.

Ce portefeuille est un *stock*, alors que les projets forment un *flux*¹⁶. Il ne convient pas de faire évoluer le portefeuille trop souvent¹⁷ et chaque investissement doit être précédé d'une évaluation de sa rentabilité mettant en regard le coût complet (y compris les dépenses de la MOA et le coût d'exploitation qu'il faudra supporter une fois le projet réalisé) ainsi que ses effets attendus.

Comme pour un portefeuille financier, il convient d'évaluer les synergies entre projets : la solidité d'un système d'information vient du fait que les applications s'appuient l'une sur l'autre comme les parties d'une voûte. Le mot « système », dans l'expression « système d'information », indique qu'il s'agit de tirer parti de ces synergies.

Quand le projet est lancé, la MOA doit modéliser le processus concerné (« spécifications générales ») dans des termes assez précis pour que l'informatique puisse se mettre au travail. L'informatique établit ensuite des spécifications détaillées que la MOA doit valider, puis les spécifications techniques qui serviront à la réalisation (voir la figure 12.5 page 419).

Pendant la réalisation la MOA dirige le projet avec l'assistance de la MOE. Si le projet dérape, si durant sa réalisation les exigences fonctionnelles évoluent, si le MOE ne remplit pas bien sa mission, c'est à la MOA qu'il reviendra de prendre les décisions nécessaires.

La MOA est toujours responsable lorsqu'un projet échoue : soit elle aura été imprécise ou versatile dans l'expression des besoins, soit elle aura manqué de vigilance face à une MOE défaillante. La MOA qui dit « c'est la faute de l'informatique » reconnaît implicitement qu'elle n'a pas correctement assumé ses responsabilités, et en outre elle a le tort de vouloir se défausser sur la MOE.

Il faut ajouter toutefois que les entreprises donnent rarement à la MOA les moyens correspondant à ses responsabilités ; elles ont tendance à croire que « tout ça, c'est de l'informatique » et à confier la totalité des responsabilités à la MOE, qui focalisera bien naturellement son attention sur les aspects techniques, physiques de la réalisation et de l'exploitation : il ne faut pas s'étonner alors si le produit, une fois livré, ne convient pas aux utilisateurs.

À la fin de la réalisation, la MOA vérifie que le produit rendra aux utilisateurs les services attendus (« recette fonctionnelle »). Elle doit en-

16. C'est pourquoi l'expression « portefeuille de projets » est malencontreuse même si elle est d'utilisation courante.

17. Les MOA sont parfois tentées de multiplier les projets. Ces innovations déstabilisent les utilisateurs.

suite assurer la formation des utilisateurs, le déploiement du produit, la « conduite du changement », l'animation des bonnes pratiques, l'écoute des réclamations et suggestions éventuelles etc.

La maîtrise d'ouvrage et les utilisateurs

Lorsqu'on parle des utilisateurs du système d'information il faut distinguer les *utilisateurs finals* qui se serviront du système d'information pour leur travail quotidien, et les *concepteurs*, généralement affectés à la direction générale et chargés de concevoir les applications en fonction des orientations stratégiques des métiers de l'entreprise.

Les concepteurs ne sont pas des utilisateurs finals mais ils sont chargés de transcrire les besoins de l'entreprise en termes de fonctionnalités à fournir par l'application.

Les relations entre maîtrise d'ouvrage et utilisateurs comportent des étapes que l'on peut énumérer dans l'ordre chronologique :

- l'expression des besoins,
- la recette fonctionnelle, qui permet aux utilisateurs de vérifier si l'application correspond bien à leur demande,
- la formation des utilisateurs finals,
- le déploiement,
- la conduite du changement,
- l'aide aux utilisateurs,
- l'exploitation technique de proximité.

L'expression de besoins

L'expression de besoins se fait en plusieurs étapes que nous allons parcourir l'une après l'autre :

- une première expression, dite « informelle » (mais néanmoins rigoureuse), permet de comprendre de quoi il s'agit. Rédigée en langage naturel, elle est compréhensible par tout le monde y compris par les dirigeants. Ceux-ci doivent la valider de telle sorte que l'on soit sûr qu'elle correspond aux priorités et à la stratégie de l'entreprise. Sans cette validation, le projet courrait le risque d'être désavoué après que l'on ait travaillé pendant plusieurs mois selon une orientation erronée ;

- on établit ensuite une expression de besoin « formalisée » dans un langage que nous allons décrire ci-dessous. La formalisation peut faire apparaître certaines lacunes de l'expression informelle : il faut itérer jusqu'à ce que l'on dispose d'une expression informelle et d'une expression formelle mutuellement cohérentes ;

- jusqu'ici le travail a été fait par la maîtrise d'ouvrage, responsable de l'expression des besoins. L'expression de besoins formalisée est communiquée à la maîtrise d'œuvre ; il faut s'assurer que celle-ci la comprenne bien. Un échange de questions et réponses conduit à l'expression des besoins définitive, dûment comprise et validée par la maîtrise d'œuvre.

Première expression des besoins

Pour rédiger la première expression de besoins, informelle, il est utile de consulter des personnes du terrain et les concepteurs. Les informations qu'apportent ces deux catégories de personnes ne sont pas de même nature : les personnes du terrain indiquent dans le détail comment les choses se passent, ce qui permet d'éviter des erreurs de conception pratique. Les concepteurs indiquent comment les choses devraient se passer, ou comment elles devront évoluer, ce qui permet d'éviter des erreurs concernant le positionnement et les orientations de l'entreprise.

La consultation des experts, les entretiens, les séances de validation, la rédaction et la vérification des comptes rendus représentent une lourde tâche matérielle. La logistique des recueils d'expertise, consultations et validations, est le talon d'Achille de la maîtrise d'ouvrage : l'expression des besoins peut échouer, en qualité ou en délai, si cette logistique n'est pas convenablement organisée. On pense trop souvent que cela va marcher tout seul mais c'est une erreur : il ne faut pas sous-estimer les efforts nécessaires pour obtenir que des personnes compétentes se rendent disponibles, pour s'assurer de leur ponctualité et de leur assiduité aux réunions ; les comptes rendus, portant sur des questions précises et parfois délicates, sont difficiles à rédiger.

On mobilise parfois des « clubs d'utilisateurs » pour produire l'expression des besoins. L'expérience incite à se méfier de cette formule même si elle a une allure « démocratique ». En rassemblant des utilisateurs on les incite en effet à exprimer une demande inflationniste, chacun se creusant la tête pour trouver une chose de plus à demander. Certains informaticiens manipulent les clubs d'utilisateurs pour faire passer leurs propres préférences : les utilisateurs demandant « tout et le reste », l'informaticien pourra ne retenir de ce fatras que ce qu'il avait envie de faire.

Pour amener un club d'utilisateurs à travailler raisonnablement, la règle d'or est de lui demander ses priorités, d'exiger qu'il indique ce qui lui paraît absolument indispensable. On ne retiendra, pour faire la « version 1 », que cet indispensable ; souvent il ne sera pas nécessaire d'aller plus loin.

La première expression de besoin devra rester, tout au long du projet, un document de référence auquel on se reporte dans les épisodes où les difficultés de la réalisation accaparent l'attention et font perdre de vue pourquoi l'on travaille.

Formalisation du besoin

Une fois l'expression de besoin rédigée et validée, il faut la modéliser de façon formelle. On dispose maintenant avec UML (*Unified Modeling Language*) d'un langage bien adapté. Il permet d'indiquer sans ambiguïté les besoins au maître d'œuvre. Il utilise les concepts propres à la programmation par objets (classes, composants, attributs, associations etc.) ; proche de l'utilisateur, dont il structure la demande, il fournit à la réalisation une base conceptuelle qui sera réutilisée et précisée lors des étapes ultérieures, la démarche suivant ainsi un processus cumulatif qui économise du travail et fait gagner du temps. Alors que les expressions de besoins étaient souvent imprécises et versatiles, le maître d'œuvre peut avec un modèle UML dispo-

ser de descriptions complètes établies selon une procédure qui garantit leur pérennité. Le modèle devient le langage dans lequel le métier structure et décrit ses fondations conceptuelles. Il sert aussi (ou plutôt il servira, car nous sommes ici en avance par rapport à la pratique de beaucoup d'entreprises) à mettre en forme les choix stratégiques et à les expliciter.

Validation du besoin par le MOAS

Les spécifications générales (ou encore « fonctionnelles ») d'un système d'information sont un document que le maître d'ouvrage fournit au maître d'œuvre et qui indique à celui-ci ce que devra faire l'outil informatique qu'on lui demande de réaliser. Ces spécifications prennent en principe aujourd'hui la forme d'un modèle UML.

Il faudra vérifier que la représentation du métier que fournit le modèle est conforme à la stratégie de l'entreprise, à son positionnement sur le marché, à ses méthodes de gestion, à ses projets, bref à sa *personnalité*.

Cette vérification-là ne peut être faite que par ceux qui définissent, qui orientent la personnalité de l'entreprise, c'est-à-dire par les dirigeants (président-directeur général, directeur général, DGA, directeurs etc.). C'est cette vérification que l'on appelle « validation ».

Cependant les choses se passent souvent de la façon suivante :

- le modèle prend la forme d'un classeur de quelques centaines de pages, hérissé de graphiques et d'acronymes, rédigé dans le jargon particulier à l'informatique (où tout texte devient un « livrable », tout donneur d'ordre un « commanditaire », toute méthode une « méthodologie », tout problème une « problématique » etc.). Le décideur y trouve divers types de diagrammes : de classes, de cas d'utilisation, de séquences, d'interaction, d'activité, d'état, de composants, de déploiement etc. Ni les cases, ni les flèches de ces diagrammes ne peuvent être compris par quelqu'un qui n'en a pas l'habitude ;

- on donne le modèle au dirigeant le vendredi soir en lui demandant de le valider pour le lundi : on est en retard, il faut qu'il signe vite... et les techniciens se disent *in petto* « tu es un dirigeant, tu l'as voulu, Georges Dandin ! Tâche de comprendre et bon week-end ! » ;

- le dirigeant signe, parce qu'il ne veut pas être celui qui ralentirait la procédure et qu'il ne veut pas avouer qu'il n'y comprend rien. Tout au plus posera-t-il une question de détail pour faire preuve de vigilance ;

- les équipes se mettent au travail en se fondant sur les spécifications ainsi « validées » ; elles passent aux spécifications détaillées, aux spécifications techniques, à la réalisation, aux tests, à la recette, au site pilote, au déploiement ... et le dirigeant constate, quelques mois après sa « validation », que le produit n'est pas conforme à la stratégie de l'entreprise. Alors il faudra soit admettre que la stratégie soit mal outillée, soit tout refaire à grands frais.

Plusieurs choses sont choquantes dans ce scénario si fréquent. Tout d'abord, le mépris avec lequel les techniciens traitent le dirigeant. Ce mépris ne s'exprime certes pas par des paroles insultantes : il peut faire bon ménage avec l'obséquiosité. Mais l'une des pires insultes que l'on puisse adresser à quelqu'un, c'est de lui donner un travail impossible en le mettant au défi de révéler son incompétence.

Puis la faiblesse du dirigeant qui accepte de galvauder sa signature en l'apposant sur des documents qu'il n'aura pas pu comprendre. Enfin la conception de l'entreprise qui sous-tend ce scénario : la validation, lorsqu'elle est de pure forme, ne représente rien de sincère et n'engage personne.

* *

La validation ne pourra être *authentique* que si elle porte sur un document lisible et compréhensible pour le dirigeant, et que celui-ci puisse le faire corriger s'il lui semble ne pas correspondre à la stratégie de l'entreprise : c'est sur ce dernier point que réside la valeur ajoutée qu'apporte la validation. Alors le dirigeant se saura engagé par sa signature ; le risque d'une inadaptation à la stratégie sera réduit ainsi que celui d'un désaveu ultérieur de la validation.

Il faut donc trouver le moyen, lorsqu'on a en mains un modèle UML, de le transcrire en un texte qui le représente fidèlement tout en étant lisible par un dirigeant. L'exercice n'est pas facile mais il est possible. On peut procéder de la façon suivante :

- rédiger une synthèse en langage naturel de quatre ou cinq pages, expliquant ce que l'on a voulu modéliser, la façon dont on s'y est pris, les modifications que l'on a apportées au processus en le modélisant, les choix que l'on a fait parmi les solutions possibles, les questions qui ont été laissées pendantes par souci de simplicité¹⁸ ;

- faire valider la fidélité de cette synthèse par quelqu'un qui connaît bien le modèle, le « vérificateur » ; il vaut mieux que ce vérificateur ne soit pas le responsable du modèle car il est difficile, pour celui qui a supervisé les détails d'une production technique, d'en vérifier une présentation synthétique ;

- présenter les cases sur lesquelles doivent figurer les signatures comme sur la figure 12.2.

Le « modèle », ce sera alors l'ensemble constitué par le modèle formel et sa synthèse. Le modèle formel est signé par le responsable du modèle et par le vérificateur ; la synthèse est signée par le vérificateur et par le dirigeant.

On communique au dirigeant la synthèse et le modèle formel. Il pourra se reporter à ce dernier s'il veut vérifier un détail, mais il ne signe que la synthèse : c'est le seul document qu'il sera censé avoir lu et qui engagera sa responsabilité.

Convergence du modèle

L'élaboration du modèle, avec ses parties formelles et ses parties en langage naturel, se fait de façon itérative. On rédige la première expression de besoins dans un langage clair et simple ; la modélisation formelle fait apparaître les ambiguïtés et incohérences inévitables dans toute première rédaction : on les corrige, ce qui conduit à construire une deuxième version du modèle formel etc. (figure 12.3). À l'issue de ce processus, la maîtrise d'ouvrage dispose d'un modèle qu'elle peut livrer à la maîtrise d'œuvre et

18. On pourra reprendre dans cette synthèse le diagramme d'activité, qui est le plus facile à lire parmi les diagrammes que comporte le modèle, et lui annexer des images d'écran qui illustreront utilement ce que l'on entend faire.

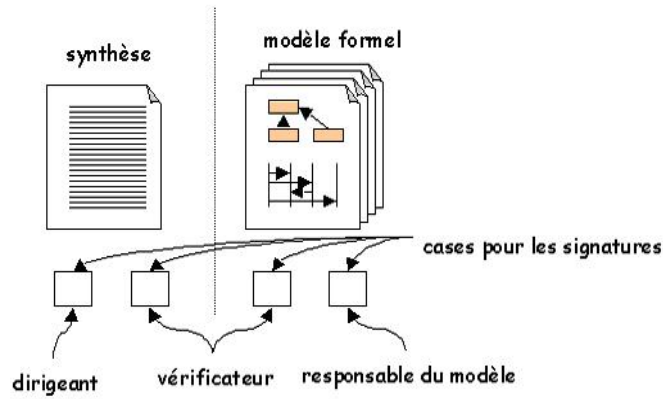


FIG. 12.2 – Modèle présenté à la validation

dont les deux parties (formelle et en langage naturel) sont mutuellement cohérentes. Avant la livraison du modèle à la maîtrise d’œuvre, les documents en langage naturel doivent avoir été validés par le MOAS.

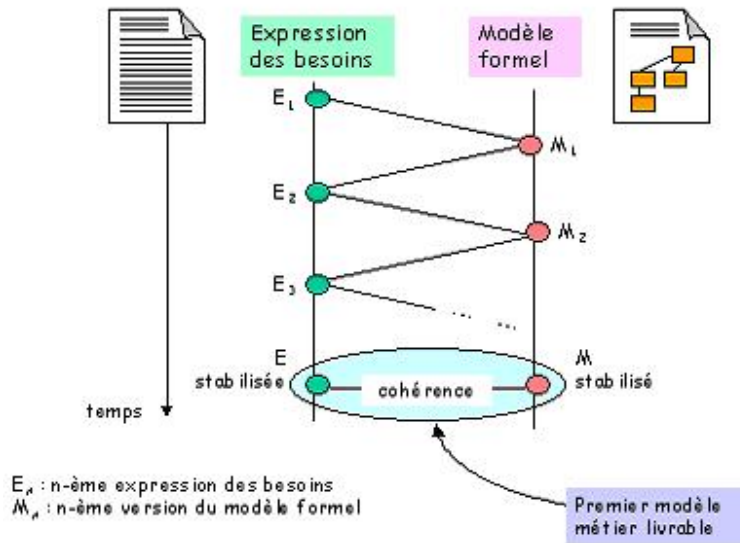


FIG. 12.3 – Convergence entre l’expression de besoins et le modèle formel

Cette élaboration demande une gestion documentaire attentive : il importe que les diverses versions soient identifiées et leur cohérence garantie de sorte que le destinataire n’ait pas à vérifier la cohérence de ce qui lui est livré.

Le livrable fourni par la maîtrise d’ouvrage à la maîtrise d’œuvre s’appelle « modèle métier », « modèle fonctionnel » ou encore « spécifications

générales » ; ces divers termes étant synonymes nous utiliserons ici l'expression « modèle métier ».

Lorsque le modèle métier est fourni au maître d'œuvre, celui-ci doit se l'approprier et s'assurer qu'il l'a bien compris. Il peut ainsi relever des points obscurs. On entre alors dans un cycle de remarques que le maître d'œuvre adresse au maître d'ouvrage, auxquelles celui-ci répond en précisant et adaptant le modèle (figure 12.4).

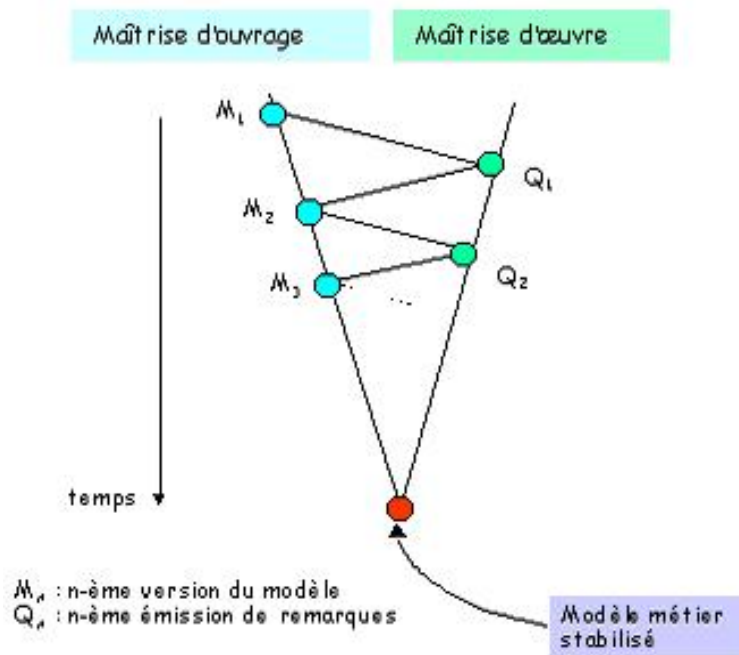


FIG. 12.4 – Convergence vers le modèle métier stabilisé

À la fin de ce cycle, on dispose d'un modèle métier stabilisé, bien compris par les deux parties et qui servira de fondement à la réalisation.

Suite de l'histoire

Nous allons maintenant examiner la démarche qui conduit à la réalisation. Le vocabulaire comporte des synonymes, mais la démarche elle-même est claire :

- le « modèle métier » (ou « spécifications générales ») est celui qui est livré par la maîtrise d'ouvrage à la maîtrise d'œuvre. C'est le « modèle métier stabilisé » décrit ci-dessus. L'expression de besoins, qui est ici notre objet, se termine lorsqu'il est établi ;

- un « modèle d'analyse » (ou « spécifications détaillées ») est ensuite rédigé par le maître d'œuvre, puis validé par le maître d'ouvrage. Il a pour but d'apporter au modèle métier des précisions techniques (cardinalité des liens, définition des classes etc.) nécessaires à la maîtrise d'œuvre. Il doit être validé par la maîtrise d'ouvrage, après quoi il sera le modèle sur le-

quel fournisseur et client se sont mis d'accord et qui servira de charte à la réalisation ;

- le « modèle technique » (ou « spécifications techniques ») est établi ensuite par la maîtrise d'œuvre qui le fournira aux programmeurs pour la réalisation proprement dite. Ce modèle, qui précise des choix techniques, reste interne à la réalisation et n'a pas en principe à être validé par le maître d'ouvrage.

Pour comprendre cette succession, prenons une métaphore inspirée de la vie courante. Supposons que vous fassiez construire une maison. Vous avez le plan sous les yeux et vous dites : « dans cette chambre, il faudra quatre prises de courant, un interrupteur commandant une prise, et une applique commandée par un interrupteur » : ce sont vos *spécifications générales*. L'électricien vous demandera ensuite de lui dire où il doit installer les prises, les interrupteurs et l'applique. Marquer sur le plan ces emplacements précis, c'est établir les *spécifications détaillées*. Puis l'électricien fera le plan de câblage qui précise le parcours des goulottes et saignées. Ce sont les *spécifications techniques*. Elles n'intéressent pas le client (mais l'électricien lui demandera peut-être son accord sur l'emplacement de l'armoire de raccordement).

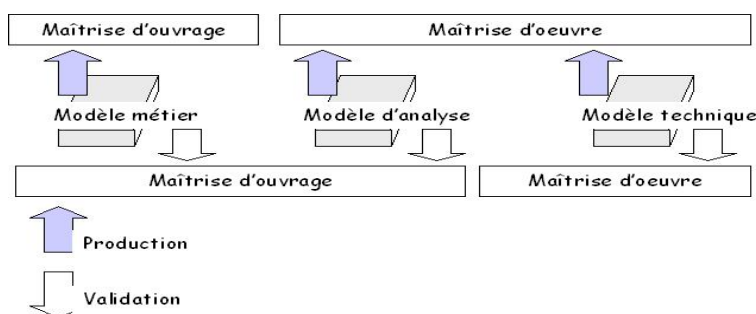


FIG. 12.5 – Partage des responsabilités entre MOA et MOE

Toute réalisation doit parcourir ces trois étapes, et être conduite de telle sorte que l'on n'ait pratiquement jamais à mettre en cause les choix opérés lors des étapes précédentes.

La maîtrise d'ouvrage est responsable à la fois de la production et de la validation du modèle métier ; pour le modèle d'analyse, la responsabilité est partagée : production par la maîtrise d'œuvre, validation par la maîtrise d'ouvrage. Enfin, la maîtrise d'œuvre est responsable à la fois de la production et de la validation du modèle technique (figure 12.5). Le découpage des rôles doit être clair dans l'esprit de chacun. Il ne signifie pas qu'il existe une cloison étanche entre maîtrise d'ouvrage et maîtrise d'œuvre : les flux d'information et les consultations doivent être réguliers tout au long de la démarche.

À la fin du processus ci-dessus, et une fois le modèle technique élaboré, on entre dans la phase de réalisation qui sera suivie du déploiement, de la formation des utilisateurs etc.

Quelques pièges

- *on sous-estime la difficulté de la logistique de consultation et de validation auprès des experts du métier* : les rendez-vous sont difficiles à obtenir, les personnes ne sont pas assidues, ou bien elles ne se sentent pas autorisées à donner un avis parce que leur mandat n'est pas clair, ou encore elles sont désavouées après l'avoir donné etc. Les délais peuvent s'allonger démesurément et la qualité de l'expression des besoins sera douteuse ;

- *on présente aux dirigeants des documents d'une technicité qui ne correspond ni à leur langage, ni à leurs préoccupations* : dès lors la validation prend beaucoup de temps, ou bien au contraire elle est rapide mais superficielle et sera remise en cause par la suite : un dirigeant ne pourra en effet jamais tolérer que l'application ne soit pas conforme à la stratégie de l'entreprise ;

- *on prend en compte les contraintes techniques de façon trop précoce* : nous avons vu ci-dessus la succession *modèle métier* → *modèle d'analyse* → *modèle technique*. Lorsqu'on construit le modèle métier, l'objectif est de donner une bonne expression de besoin et non d'optimiser les solutions techniques qui devront être examinées ensuite. Mais il arrive parfois que sous prétexte de « sérieux », de « rigueur » ou de « méthodologie » un consultant ou un informaticien impose au métier des règles de modélisation qui anticipent sur les choix à réaliser dans le modèle d'analyse ou même dans le modèle technique. Ces choix précoces (sur la structuration des composants, les interfaces etc.) devront ensuite être révisés par le maître d'œuvre, d'où travail en double et perte de temps ;

- *la cloison entre maîtrise d'ouvrage et maîtrise d'œuvre est étanche* : même si la maîtrise d'œuvre n'est pas responsable du modèle métier, il est utile que le maître d'ouvrage la consulte pour s'assurer de la faisabilité de ce qu'il envisage ; par ailleurs, même si le modèle technique ne concerne pas le maître d'ouvrage, il est utile que celui-ci soit informé de choix qui auront pour lui des conséquences en termes de performances, fiabilité etc. : la spécialisation des rôles, la séparation des responsabilités ne doivent pas exclure la collaboration.

12.2.4 Rôle de la maîtrise d'œuvre

Nota Bene : La description du rôle de la MOE est ici succincte, nous y reviendrons page 463.

La MOE assiste la MOA lors de l'évaluation du coût d'un projet, car cette évaluation suppose que l'on dispose au moins d'une esquisse de solution technique. Elle établit les spécifications détaillées que la MOA valide ; elle établit enfin les spécifications techniques, dernière étape avant la réalisation. La MOE rédige alors les cahiers des charges, lance les appels d'offre, négocie les contrats de réalisation avec les entreprises (notamment avec l'entreprise chargée de l'intégration de la solution).

Le chef de projet MOE assure la conduite du chantier, sous la direction de la MOA à laquelle il fournit un suivi du planning de réalisation ainsi que toute indication technique utile à la direction de projet. La MOE valide

les recettes techniques (« recettes usine ») faites par les fournisseurs. Elle assiste ensuite la MOA lors de la recette fonctionnelle. Enfin, elle assure la mise en exploitation de la solution sur la plate-forme technique, puis son exploitation continue, en garantissant la qualité du service (disponibilité, fiabilité etc.) ainsi que le support aux utilisateurs.

12.3 Articuler l'expertise et la décision

L'un des principes essentiels de l'organisation de l'entreprise est de *séparer l'expertise et la décision*. Beaucoup d'entreprises cultivent l'illusion selon laquelle le dirigeant serait un expert. Certes ce n'est pas un ignorant : il a souvent été, dans une vie antérieure, un spécialiste compétent, un expert dans un domaine professionnel. Mais lorsqu'il accède à une fonction de direction son expertise s'efface. Il devra prendre des décisions concernant l'ensemble de l'entreprise et dont la portée embrasse celle de plusieurs spécialités. Même quand la décision relève de son ancienne spécialité, il ne pourra pas tirer parti de son ancienne expertise : il n'a pas pu l'entretenir, il n'a pas pu suivre l'état de l'art. Rien n'est plus dangereux qu'un dirigeant qui, expert naguère, imposerait des choix techniques désuets.

Il est donc préférable de postuler que le dirigeant n'est pas un expert, qu'il n'est pas en mesure d'instruire à fond les choix qui se présentent à l'entreprise. Cette hypothèse est contraire à certaines habitudes : l'entreprise croit parfois devoir, malgré l'évidence, attribuer au dirigeant une lucidité, une science hors de la portée de l'être humain ordinaire ; elle suppose qu'il bénéficie, comme le Pape, d'une grâce d'état qui conférerait l'infaillibilité à ses décisions. Cette conception est, dans l'entreprise, l'un des héritages de l'organisation ecclésiastique.

Que doit attendre l'entreprise de son dirigeant, quelle est la valeur ajoutée propre de celui-ci ? Elle réside d'abord dans la légitimité de la décision, dans le pouvoir d'arbitrage qui permet de trancher les hésitations de la réflexion, la concurrence entre les ambitions, et qui indique l'action à laquelle toutes les énergies devront se consacrer. Elle réside aussi dans la vigilance périscopique qui, survolant les détails de l'exécution et surveillant l'environnement, apporte à l'entreprise autant de sécurité qu'il est possible dans un univers incertain. Elle se concentre enfin sur les enjeux essentiels : positionnement sur le marché ; relations avec les partenaires, fournisseurs, créanciers ; qualité de l'organisation et du système d'information.

Que doit attendre l'entreprise d'un expert ? Qu'il assure une veille sur son domaine afin de se tenir au courant de l'état de l'art (qualité et coût des diverses solutions) ; qu'il soit en mesure, quand l'entreprise se pose une question relevant de son domaine, de lui apporter dans un court délai l'éclairage nécessaire. L'expert, notons-le, travaille d'autant mieux qu'il est soulagé de l'angoisse de la décision ; il apporte des indications au décideur mais ne prend pas la décision lui-même¹⁹.

19. Lors de l'affaire de la vache folle, des experts furent consultés pour éclairer la décision sur les importations de viande venant de Grande-Bretagne. Lionel Jospin déclara qu'il

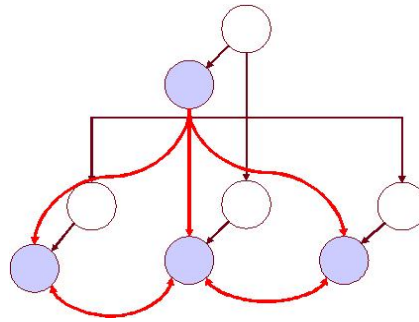


FIG. 12.6 – Réseau des MOAD

Si la séparation de l'expertise et de la décision est pour l'entreprise une règle générale, elle s'applique de façon particulièrement impérieuse dans le domaine du système d'information où l'évolution technique est rapide et où la majorité des dirigeants n'ont jamais eu de compétence.

C'est pourquoi il est recommandé d'adjoindre à chaque dirigeant, maître d'ouvrage stratégique dans son domaine (MOAS), un maître d'ouvrage délégué (MOAD) qui remplira auprès de lui une fonction d'expertise. Dans la figure 12.6, les ronds placés au dessus désignent les dirigeants (ou MOAS), les ronds placés au dessous les MOAD ; l'organigramme, représenté par les flèches droites, indique les relations entre le directeur général, dirigeant suprême, et les DGA ou directeurs responsables des divers métiers. Les flèches courbes désignent des relations entre MOAD que nous commentons ci-dessous.

Nous allons décrire l'organisation idéale, qui se rencontre sous une forme plus ou moins exacte dans les entreprises les plus avancées en matière de système d'information (en particulier dans le secteur des banques et assurances où les entreprises, luttant pour la survie, ont pris quelques années d'avance).

Le MOAD doit veiller à l'urbanisme du système d'information et à l'appropriation de cet urbanisme par le MOAS. Cela suppose que le MOAD présente la nature et les enjeux du système d'information dans un langage que le MOAS puisse comprendre : ce langage doit donc être « sur mesure », finement adapté à la personnalité de ce MOAS.

Parmi les MOAD existe une relation fonctionnelle. L'un des MOAD assiste le directeur général lui-même et prépare les arbitrages définitifs. Il se trouve donc en position de coordinateur par rapport aux autres MOAD : il lui revient d'animer leur réseau, de veiller à l'utilisation de méthodes convenables, de préparer l'interclassement des priorités entre les demandes que présentent les diverses directions.

suivrait leur avis : cela mit les experts mal à l'aise, car ils sentaient la diversité des dimensions que la décision devait prendre en compte et que leur expertise ne pouvait pas couvrir.

Enfin les MOAD échangent des informations qui contribuent à aplanir les barrières entre les directions et à promouvoir des normes communes (par exemple par le respect du référentiel de l'entreprise). Cette collaboration transverse est particulièrement opportune dans les entreprises qui veulent utiliser de façon cohérente divers médias (Internet, téléphone, courrier etc.) dans les relations avec leurs clients, fournisseurs et partenaires.

La direction des systèmes d'information (DSI), responsable de l'informatique, est en général chargée de la maîtrise d'œuvre du service informatique qu'utilisent les autres directions ; elle est aussi maître d'ouvrage de la plateforme informatique, qu'elle a pour mission de faire fonctionner. Ainsi le DSI est lui aussi un MOAS, et il convient qu'il soit assisté par un MOAD (ce sera souvent son directeur de l'architecture).

Les arbitrages ultimes concernant le système d'information, notamment ceux relatifs au budget annuel, sont pris par le comité de direction de l'entreprise réuni selon une configuration spéciale, le CSSI, « Comité Stratégique du Système d'Information ».

L'ordre du jour des réunions de ce comité doit être préparé conjointement par le MOAD qui assiste le directeur général et le MOAD qui assiste le DSI de façon à garantir l'équilibre entre les priorités techniques de l'informatique (dimensionnement des ressources, choix des solutions) et les priorités fonctionnelles des métiers.

Le CSSI rassemble les divers MOAS, qui sont là pour défendre les propositions de leurs directions respectives et participer collégialement à la prise de décision, et leurs MOAD qui sont là pour assister chacun le MOAS auprès duquel il joue ainsi le rôle d'un *sherpa*.

En principe, seuls les MOAS ont droit à la parole autour de la table du CSSI, les MOAD se tenant en retrait comme le fait le conseiller d'un ministre en réunion interministérielle ou au Parlement. En pratique, les choses se passent le plus souvent de façon détendue et informelle, étant entendu que chacun pèse dans la discussion selon la légitimité que lui confère sa fonction.

12.4 Compétences de la maîtrise d'ouvrage

Un MOAD doit savoir définir des priorités, urbaniser, modéliser, obtenir des validations authentiques, animer les bons usages. Cela suppose qu'il sache gérer le nœud des relations qu'il entretient avec les utilisateurs, le MOAS et la maîtrise d'œuvre, et qu'il sache parler à chacun dans le langage qui convient.

Vis-à-vis de l'informatique, en particulier, il doit savoir se comporter en *client compétent*. L'image qui vient ici à l'esprit est celle d'une dame élégante en conversation avec son couturier : elle n'a peut-être jamais de sa vie tiré une aiguille mais elle sait parler le langage du métier ; elle connaît les tissus, s'intéresse à la coupe ; elle sait distinguer la chaîne et la trame, dire si ça grigne ou si ça gode, indiquer où placer les pinces... De même, le MOAD

n'aura peut-être jamais écrit une ligne de programme²⁰. Il ne lui revient ni de définir l'architecture technique, ni de programmer ; il ne prétendra jamais être un informaticien mais il doit s'intéresser assez à l'informatique pour comprendre ce que dit l'informaticien : il faut qu'il en sache en informatique ni trop, ni trop peu.

On doit distinguer deux phases dans la coopération entre maîtrise d'ouvrage et maîtrise d'œuvre :

- *dialogue* : aussi détendu que possible et aussi complet qu'il est nécessaire. Chacun est libre de donner son avis sur les questions qui relèvent de la responsabilité de l'autre ;

- *la décision* : chacun rentre dans sa tente pour décider dans son domaine propre, et s'interdit d'empiéter sur les responsabilités de l'autre.

Il serait dommage d'interdire à un chef de projet MOE, qui connaît bien le métier pour lequel il travaille depuis des années, de formuler des suggestions lors de la définition des fonctionnalités. Il serait de même dommage d'interdire à un MOAD, qui utilise l'informatique depuis des années, de formuler des suggestions sur la solution informatique. Mais la discussion sera d'autant plus libre, d'autant plus détendue, qu'il est bien entendu que chacun sera, *in fine*, le seul à détenir le pouvoir de décision légitime dans son domaine de responsabilité propre.

Certains des problèmes que doit traiter le chef de projet MOE, comme les spécifications techniques ou les contraintes relatives à la plate-forme technique elle-même, ne présentent d'ailleurs aucun intérêt pour un MOAD.

Ajoutons cependant que la MOA, client de l'informatique, a le devoir de refuser une solution qui lui semblerait trop coûteuse, trop risquée, trop longue à réaliser ou incorrecte au plan fonctionnel. En modifiant le slogan de mai 68, on peut dire aux MOAD « soyez réalistes, demandez tout le possible (mais rien que le possible) ».

Nous allons parcourir du bas en haut les diverses couches que comporte la plate-forme technique d'un système d'information et suggérer la liste de ce que devrait savoir un MOAD en informatique. Par « savoir », nous n'entendons pas dire ici que le MOAD doit connaître exactement la façon dont fonctionne chacun des outils que nous énumérons (ce serait là l'expertise d'un informaticien), mais qu'il doit connaître leur existence et avoir une idée juste des services qu'ils rendent.

Poste de travail

Le poste de travail est trop souvent jugé indigne de retenir l'attention. Pourtant dans les entreprises le PC en réseau est l'outil quotidien de chacun (voir page 201). Il permet d'accéder au système d'information, de le consulter, de saisir des données et de lancer les traitements. Son achat, son installation, son entretien, l'aide à l'utilisateur représentent parfois plus de la moitié du budget informatique (voir page 529).

20. La programmation est cependant pour le MOAD un *hobby* salubre, car il l'aide à entrevoir les difficultés que rencontre l'informatique. Mais il restera toujours un écart important entre la programmation personnelle, qui est du bricolage, et la programmation telle qu'elle se pratique dans les grandes usines que sont les SSII.

Il mérite donc qu'on lui consacre *beaucoup d'attention*. Le MOAD doit connaître le PC en réseau et son système d'exploitation (Windows est le plus courant, mais il va falloir aussi considérer Linux). Il doit connaître le protocole du réseau local (Ethernet, FDDI et les protocoles utilisés dans le WAN (*Wide Area Network*) qui relie le réseau local au reste de l'entreprise (TCP/IP, X25, Frame Relay, ATM etc.) ainsi que la diversité des accès au réseau dont dispose un établissement (RTC, Numéris, liaisons louées, ADSL etc.). Les options en matière de réseau (débit disponible, gestion des priorités, sécurité) ont en effet des incidences sur la qualité du service. Le MOAD doit savoir comment se répartissent les tâches entre le poste de travail, le serveur local et les serveurs centraux (*mainframes*, serveurs Unix et Wintel, *proxys*).

Services

L'utilisateur dispose sur son poste de travail d'une panoplie d'outils bureautiques (traitement de texte, tableur, logiciel graphique) et de bureautique communicante (messagerie, agenda partagé, documentation électronique, workflows, forums) qui s'ajoutent à ses accès aux applications centrales. Il faut s'assurer que l'utilisateur ait les compétences nécessaires pour tirer parti de ces fonctionnalités.

L'infrastructure du système d'information fournit des services qui outillent toutes les applications : annuaire des personnes, des équipements etc. (voir page 371), SGBD, impression. Les annuaires contiennent les identifiants (de l'organisation, des personnes, des équipements) et une sélection d'attributs qui les décrit. L'annuaire des personnes est utilisé pour la messagerie, la téléphonie, et fournit les profils nécessaires à l'habilitation. Les SGBD (Oracle, SQL Server de Microsoft, Informix d'IBM etc.) assurent le classement et la mise à disposition des données. Les services d'impression²¹ sont répartis entre l'impression locale (courrier ordinaire) et des services d'impression centrale (courrier de masse).

Infrastructure de communication

L'infrastructure de communication (*middleware*) réalise les échanges de données entre les composants et assure l'interopérabilité des systèmes d'information d'entreprises différentes (entreprise étendue). Elle doit donc traiter des problèmes d'ordre physique et logique :

1) le message émis par un composant doit être routé vers le composant destinataire. On peut utiliser des « brokers » ou des EAI (voir page 294) pour gérer les tables d'adressage, ce qui simplifie d'autant la définition des composants ;

2) si le codage des données n'est pas le même dans deux applications, il faut un transcodage pour les faire communiquer ; cela peut impliquer une perte de qualité des données. Le transcodage automatique est approximatif et il faudra un traitement manuel des cas douteux ;

3) synchronisme : on ne peut pas faire communiquer sans précautions des applications non synchronisées. En allant du plus rapide au moins rapide,

²¹. La qualité des services d'impression est dans beaucoup d'entreprises un point faible mais trop négligé.

donc du plus coûteux au moins coûteux, on distingue la communication en mode transactionnel, immédiate (Tuxedo de BEA) ; la communication en mode message, rapide mais soumise à l'aléa des files d'attente (MQSeries d'IBM) ; la communication en batch (traitement différé au soir ou au week-end) ;

4) volumétrie : la solution dépend du volume à échanger. Si le canal est trop étroit, les files d'attente débordent et des messages sont perdus ;

5) traitement des rejets : les messages que l'interface entre deux applications refuse seront traités le plus souvent de façon mi-automatique, mi-manuelle ;

6) concurrence : lorsque plusieurs personnes ont accès en même temps au même dossier il faut faire en sorte que leurs actions n'entrent pas en conflit ;

7) Les industriels du logiciel offrent des plates-formes de développement et d'exploitation qui facilitent la mise en place, l'administration et l'exploitation des composants, et assurent aussi certaines des fonctions de l'infrastructure de communication évoquées ci-dessus (Websphere d'IBM).

Des outils de groupware (Lotus Domino, Microsoft Exchange) permettent de configurer la documentation électronique, de gérer les droits des utilisateurs, de construire de petits workflows (masques de saisie et d'interrogation, traçabilité), d'outiller la messagerie en outils de classement, tri, filtres, etc. ; ils permettent la rédaction coopérative ainsi que la dissémination sélective. Des outils spécialisés permettent de réaliser des workflows plus complexes.

Applications

Le MOAD doit connaître et utiliser les outils de modélisation UML (All-Fusion Component Modeler de Computer Associates²², Power AMC 9.0 de Sybase etc.). La présentation des modèles à la validation par les MOAS s'appuiera utilement sur des outils de visualisation et médiatisation (OnMap de Nomia). Il faut aussi, même s'il ne développe pas lui-même, que le MOAD connaisse les outils utilisés pour la réalisation des produits (PowerBuilder de Sybase ou Progress AppModeller), les générateurs de code, ainsi que les outils de test et d'intégration.

Le système informatique d'aide à la décision est la fine pointe du système d'information ; il utilise des outils de datawarehouse comme SAS (regroupement et classement de données synthétiques) et, plus rarement, de datamining (recherche des données détaillées explicatives des données synthétiques).

Nous évoquerons plus loin la gestion du portefeuille applicatif (page 522), d'urbanisme (page 433), de modélisation (page 445), de conduite de projet (page 453) etc.

Le MOAD doit connaître l'offre de progiciels, qu'il s'agisse de progiciels généraux (ERP, voir page 289) ou de progiciels spécialisés par domaine (gestion de ressources humaines, gestion des achats et des stocks, comptabilité, gestion de la relation clientèle etc.) afin de pouvoir distinguer ce qui doit être fait de façon spécifique de ce que l'on peut confier à un progiciel.

22. Héritier de Paradigm Plus de Platinum.

Qualité du service

Pour administrer le système d'information, la direction informatique doit être équipée d'outils qui lui permettent d'assurer la qualité du service et facilitent les mises à jour : inventaire, supervision, télédistribution, support aux utilisateurs. Il faut que le MOAD les connaisse et soutienne à l'occasion les demandes budgétaires de l'informatique.

L'inventaire des machines (serveurs, équipements du réseau, poste de travail) doit être tenu à jour, ainsi que celui des logiciels installés sur chaque machine. Ce n'est pas une mince affaire surtout quand l'utilisateur est autorisé à charger des applications personnelles sur son poste de travail. La discipline dans la répartition des rôles entre le serveur (central ou local) et le poste de travail permet de contenir les tendances anarchiques.

Il est important que la qualité du service soit garantie du serveur central jusqu'au poste de travail de l'utilisateur²³. La supervision est, comme une tour de contrôle, le lieu où arrivent les informations et alarmes sur le fonctionnement des réseaux et des machines ; les superviseurs sont équipés d'outils permettant de reconfigurer le réseau ou de déclencher des interventions rapides sur le terrain. La mise à jour des versions logicielles sur les serveurs locaux et les postes de travail, le chargement des nouveaux logiciels, peuvent se faire « à la main » localement mais cela prend du temps et pendant le délai de mise à jour l'entreprise doit faire fonctionner en parallèle des versions différentes. La télédistribution des logiciels permet d'effectuer les mises à jour *via* le réseau et de les synchroniser.

Le *support aux utilisateurs* recouvre l'installation des postes de travail et équipements périphériques (explications orales, règlement « à chaud » des questions de paramétrage et de connectique), le traitement des pannes (explication de l'origine de la panne, indication sur sa durée, intervention du réparateur), enfin l'aide patiente et bienveillante à l'utilisateur inexpert. Des personnes physiquement proches de l'utilisateur effectuent une assistance de premier niveau (on estime qu'il faut lui consacrer de 1 à 2 % des effectifs), puis vient l'aide en réseau fournie par des centres d'appel qualifiés, enfin pour les questions les plus difficiles les centres d'appel mobilisent des experts nationaux ou des fournisseurs.

12.5 Défis de l'informatique aujourd'hui

Quelles sont les difficultés principales que les maîtrises d'ouvrage et les maîtrises d'œuvre rencontrent aujourd'hui ? Les conférences du club des maîtres d'ouvrage ont fait apparaître quelques priorités : urbanisation, évolution des techniques et du positionnement, frontières entre ce que l'entre-

23. La responsabilité de l'informatique ne se limite pas au fonctionnement du serveur central. Il arrive qu'un arrêt de quelques minutes du serveur central provoque une panne de plusieurs heures pour les utilisateurs en raison des délais de remise en route des routeurs et serveurs situés entre le poste de travail et le serveur central. La durée de panne à retenir dans ce cas pour l'indicateur de qualité, c'est bien *le nombre d'heures d'arrêt du service* et non le nombre de minutes d'arrêt du serveur central.

prise doit faire elle-même et ce qu'elle doit se procurer sur le marché. La plupart des systèmes d'information des grandes entreprises ont été construits au fur et à mesure, alors que les méthodes et techniques informatiques évoluaient ; on peut qualifier certains d'entre eux de « bidonvilles de luxe » : bidonvilles parce qu'ils sont en désordre, « de luxe » parce qu'ils coûtent cher (une entreprise du secteur tertiaire dépense pour son système d'information de 3 000 à 10 000 € par salarié et par an).

Du point de vue des maîtrises d'ouvrage la démarche d'urbanisme passe par la construction d'un référentiel qui documente les domaines, processus, activités, composants, données et aide l'entreprise à définir ses priorités pour la mise à niveau du système d'information.

Du point de vue des maîtrises d'œuvre, l'urbanisme passe par une structuration de la plate-forme technique que l'expression « Enterprise Application Integration » (EAI) désigne par un acronyme d'une simplicité fallacieuse (voir page 294). L'urbanisme vise aussi à maîtriser le coût du système d'information, notamment les coûts d'exploitation et de maintenance²⁴.

La gestion du portefeuille applicatif doit s'appuyer sur des études économiques qui éclaireront, fût-ce de façon qualitative (car les évaluations prévisionnelles sont toujours imprécises) la définition et la sélection des projets.

* *

Nos entreprises utilisent désormais une grande diversité de médias pour communiquer avec leurs clients et fournisseurs : aux médias traditionnels (face-à-face, courrier et téléphone) se sont ajoutés la messagerie, la Toile, parfois la carte à puce.

L'automatisation des traitements permet d'innover dans la relation commerciale. L'organisation de l'entreprise, son positionnement, le fonctionnement des marchés sur lesquels elle intervient en sont transformés. Il importe que le client perçoive l'unicité d'une entreprise avec laquelle ses relations empruntent divers canaux : le système d'information doit permettre à l'opérateur du centre d'appel de connaître les transactions que le client a réalisées sur l'Internet, les comptes rendus des réunions avec le client sont mis sur l'Intranet etc. La cohérence et l'enrichissement fonctionnel de la communication multimédia, que l'on désigne par le terme GRC (« Gestion de la Relation Clientèle », en anglais CRM pour *Customer Relationship Management*), imposent au système d'information une haute exigence de qualité.

Par ailleurs nos entreprises cherchent à consolider leur positionnement en enrichissant leur offre grâce à des partenariats qui permettent d'offrir au client des assemblages (*packages*) intégrant des produits offerts par plusieurs entreprises. La commercialisation d'une offre conjointe fournie « sans couture » suppose de faire interopérer des systèmes d'information. Or des systèmes d'information ne peuvent interopérer que s'ils sont de haute qualité. Une entreprise dont le système d'information est de mauvaise qualité

24. L'attention se focalise souvent sur le coût des projets, mais après un projet l'entreprise devra chaque année payer l'exploitation et la maintenance du produit : l'accumulation des coûts d'exploitation et de maintenance finit par représenter un budget nettement supérieur à celui qui est disponible pour les projets.

ne saura pas fournir à ses clients une relation multimédia satisfaisante et les partenariats dans lesquels elle s'engagera ne pourront pas fonctionner durablement.

* *

L'élargissement des possibilités offertes par l'informatique a eu pour contrepartie un accroissement de sa complexité. Elle connaît une évolution analogue à celle de la médecine au début du XX^e siècle et éclate en *spécialités* dont chacune exige de l'expert un travail à temps plein. On n'est plus informaticien tout court, mais spécialiste en sécurité, en administration de réseau, en *middleware*, en Java etc. Chacune de ces spécialités éclate encore en spécialités plus fines et il n'est pas facile pour le DSI(d'animer la coopération entre des experts relevant de tant de spécialités différentes.

Quelles sont les spécialités qu'une entreprise doit savoir maîtriser, quelles sont celles qu'il vaut mieux se procurer auprès d'un prestataire? Pour répondre à cette question, le DSI doit tracer une frontière aux contours délicats.

Dans le domaine des logiciels, il devra encore tracer une autre frontière : celle qui passe entre les logiciels spécifiques et les progiciels. Il serait stupide de programmer un logiciel spécifique pour une tâche qu'un progiciel relativement peu coûteux fait convenablement, comme par exemple le traitement de texte ; à l'autre extrémité du spectre, dans le cœur de métier de certaines entreprises, se condense une expertise très spécifique qu'aucun progiciel ne pourrait outiller. Entre ces deux situations également simples on peut classer du plus au moins spécifique les logiciels dont l'entreprise a besoin. On doit situer dans ce classement la frontière entre progiciel et spécifique, mais elle se déplacera car l'offre de progiciels évolue. La veille système d'information devra être attentive à cette évolution et au coût de mise en œuvre d'un progiciel.

Chapitre 13

Méthodes

Les méthodes, que l'on nomme parfois de façon emphatique « méthodologies », sont du bon sens en conserve, de l'intelligence surgelée. Si les cuisiniers médiocres utilisent abondamment les conserves qui les soulagent en faisant la cuisine à leur place, les bons cuisiniers n'en abusent pas : de même, les bons experts n'abuseront pas des méthodes.

Mais souvent, de façon toujours surprenante, celui qui expose un problème s'entend demander « quelle est votre méthodologie ? » - si l'on répond que l'on considère d'abord le problème avant de se soucier de méthodologie, on passe pour quelqu'un de peu sérieux...

La méthode est la meilleure et la pire des choses. Celui qui n'a aucune méthode tourne en rond et gaspille son énergie en pure perte ; ceux qui se rengorgent en parlant de méthodologie s'attachent trop souvent davantage à la lettre, qui tue, qu'à l'esprit qui seul vivifie.

En avons-nous vu, de ces classeurs remplis de formulaires illisibles mais qu'il avait fallu produire pour se conformer à la « méthodologie de démarche qualité », de ces armoires de documentation qui servaient d'alibi au fournisseur d'un produit bogué : « j'ai suivi la démarche en V », disait-il fièrement, comme si cela suffisait pour que son produit fonctionnât... Il croyait sa méthode bonne à tout comme celle des médecins dans le *Malade imaginaire* de Molière (1673) :

*Clysterium donare,
Postea seignare,
Ensuitta purgare.*

L'envie de rire se dissipe lorsqu'on se remémore l'obésité de nos systèmes d'information (80 % de fonctionnalités inutilisées mais dont il faut assurer la maintenance) et le niveau des taux d'échec (voir page 453). Certes il se produit des échecs ailleurs - des immeubles s'effondrent, des machines souffrent de malfaçons - mais dans aucun autre domaine de l'ingénierie on ne rencontre autant de sinistres que dans celui des systèmes d'information.

C'est qu'il existe un bon et un mauvais usage de la méthode. Le mauvais usage est strictement formel. On définit à l'avance, en s'appuyant sur un

standard, la liste et le plan des documents que l'on doit produire. Le papier s'empilera feuille après feuille, classeur après classeur, puis on programmera en conséquence mais on aura en cours de route perdu de vue le but et les priorités : alors l'erreur, devenue invisible, se glisse sournoisement entre les lignes de la méthode.

La meilleure façon d'utiliser une méthode quelle qu'elle soit - qu'il s'agisse de Merise, de COBIT, d'ITIL ou autres - c'est de la laisser d'abord de côté pour réfléchir posément à la question posée et établir en toute liberté une esquisse de solution. Il ne sera pas mauvais de disposer, pour cette première étape, sinon d'une méthode, du moins d'une check-list sommaire permettant de savoir par quel bout on doit aborder le problème.

C'est ensuite seulement que l'on pourra revenir à la méthode pour voir si l'on n'a rien oublié d'important puis sélectionner, dans la liste des opérations qu'elle énumère, celles qu'il est nécessaire d'effectuer - et laisser tomber les autres.

Les méthodes, faites pour répondre à une grande diversité de cas, comportent en effet beaucoup trop de règles, beaucoup trop de documents pour chaque cas particulier : il faut les élaguer tout comme on doit élaguer la demande des utilisateurs pour établir une expression de besoin. Il ne faut jamais se sentir obligé de suivre la méthode à la lettre, ni moins encore bien sûr l'utiliser comme un parapluie en se disant « si le produit ne marche pas, on ne pourra rien me reprocher puisque j'aurai suivi la méthode. »

* *

Lorsque les méthodes se concentrent sur l'évolution du système d'information, sur le choix des projets à réaliser et la conduite de projet, elles négligent le point le plus important : animer l'utilisation du système d'information existant. À quoi servent les investissements, en effet, si l'on néglige d'utiliser les outils dont on dispose ?

Il faut donc observer et connaître les utilisateurs, segmenter leur population en pratiquant un « marketing interne » (voir page 395), connaître leurs pratiques afin de redresser courtoisement celles qui sont inefficaces et d'encourager la diffusion des bonnes pratiques. L'animation du système d'information suppose de collecter l'information par voie d'enquête (par exemple une enquête de satisfaction périodique) puis de diffuser les bonnes pratiques en se servant de l'Intranet, de la messagerie et des réunions.

Il faut aussi que le système d'information soit connu des concepteurs qui préparent son évolution comme des décideurs qui doivent arbitrer entre les priorités ; c'est l'un des buts de l'urbanisation (Longépé [119]). Il faut enfin que les enjeux économiques soient convenablement évalués : le système d'information constitue un patrimoine qui doit être géré en tant que tel et, comme tout actif, il doit être judicieusement défini et correctement dimensionné. Il faut éviter que l'entreprise soit sous-informatisée (ou sur-informatisée), il faut qu'elle définisse la priorité relative de ses divers investissements : cela nous conduit au seuil de la préparation des projets.

Chaque année, les directions de l'entreprise proposent à la direction générale d'investir dans le système d'information. Chaque direction juge na-

turellement ses projets plus importants que ceux des autres : il revient au directeur général d'arbitrer.

Les MOAD contribuent à la préparation des projets de leur direction, à l'instruction des dossiers, à la définition des priorités. Le choix proposé au directeur général doit tirer parti des études qui permettent d'évaluer l'opportunité de chaque projet et de la comparer à celle des autres projets.

Les projets, épisodes très visibles dans la vie de l'entreprise, monopolisent l'attention au détriment de l'animation du système d'information existant ; c'est un peu comme si, dans une ville, la priorité résidait dans des chantiers de construction et non dans la vie quotidienne des habitants. C'est un facteur d'inflation, et il en existe d'autres : le prestige d'une direction s'évalue parfois selon l'ampleur du budget qu'elle est capable d'obtenir. La sobriété, quoi que l'on dise, n'est pas toujours à l'ordre du jour.

On utilise souvent l'expression « portefeuille de projets » pour désigner l'ensemble des projets finalement sélectionnés. Cette expression est malheureuse car le mot « portefeuille » s'applique en principe à un patrimoine et non au flux des investissements. Elle trahit l'importance excessive que l'on accorde aux projets, aux transformations du système d'information, par rapport à sa bonne utilisation.

* *

Le désir de bien faire, le souci de faire preuve de compétence, poussent parfois à réaliser des travaux trop lourds ou à anticiper sur les étapes suivantes d'un projet.

1) Ainsi l'expression de besoin, établie par la maîtrise d'ouvrage, décrit le produit nécessaire et indique pourquoi il est nécessaire, à quoi il doit servir. Elle répond aux questions « quoi, pourquoi ». La réponse à la question « comment fournira-t-on ce produit » relève non de l'expression de besoin, mais de l'étude qui vient ensuite. Or il arrive que le rédacteur d'une expression de besoin se sente obligé non seulement de décrire le besoin, mais aussi de dire comment il sera satisfait - alors qu'il n'a pas la compétence technique nécessaire et que son analyse sera souvent contredite lors de l'étude.

2) Les « spécifications générales » que comporte le cahier des charges n'ont pas vocation à régler les questions qui devront être traitées lors de la rédaction des « spécifications détaillées ». Il arrive que par souci de « sérieux » le rédacteur des spécifications générales aille trop loin dans le détail : l'échafaudage ainsi construit sera démoli par celui qui vient après et qui doit, lui, établir les spécifications détaillées.

Il faut donc avoir pour principe de faire à chaque étape ce qu'il est prévu de faire dans l'étape considérée, rien de moins et rien de plus.

L'enchaînement des tâches doit être organisé de telle sorte que les documents produits jusqu'à une étape donnée constituent un corpus documentaire cohérent et puissent être réutilisés tels quels, et enrichis, lors des étapes suivantes. La rédaction des documents sera ainsi cumulative et l'on n'aura pas à réécrire l'expression de besoins lors de l'étude OFR, l'étude OFR lors du cahier des charges, les spécifications générales lors des spécifications détaillées etc. Des corrections éventuelles devront être apportées

aux documents amont pour tenir compte des éléments nouveaux survenus lors de la réalisation.

La cohérence documentaire est un puissant facteur d'efficacité : si le projet est bien conduit, un même outil pourra être utilisé pour mettre en forme les spécifications générale, puis détaillée, puis technique, et enfin alimenter un générateur de code (par exemple Rose, de Rational) qui réalisera et documentera automatiquement une partie du logiciel.

13.1 Urbaniser un système d'information

Deutroisix¹ est en 1999 une grosse entreprise de service : 20 000 salariés tous équipés de PC en réseau, 1 000 agences, un usage intensif du système d'information, un budget informatique annuel d'environ 150 millions d'euros. Le cœur de métier de Deutroisix, c'est l'intermédiation du marché du Troisdeneuf. Lors d'un entretien avec un client, les informations sur sa demande sont stockées dans une base de données, confrontées avec des offres (« rapprochement ») et des mises en relation sont opérées.

Beaucoup de très gros projets sont menés simultanément : l'un pour l'intermédiation du marché du Troisdeneuf ; l'autre pour la gestion des ressources humaines ; un autre encore pour l'aide à la décision ; des « services à distance » avec un site sur la Toile et des centres d'appel. Par ailleurs l'informatique de communication se développe avec la mise en place de la messagerie, de l'agenda partagé et d'un Intranet qui permet des applications de gestion électronique de documents, de rédaction coopérative et de dissémination sélective. Des workflows sont en cours de conception. Deutroisix est en train de rattraper à vive allure un certain retard en matière de groupware et d'applications sur la Toile tout en mettant en place de « bonnes pratiques » chez les utilisateurs.

L'informatique est techniquement efficace (on n'entend plus parler de pannes informatiques alors que c'était une plainte fréquente voici quelques années). Cependant la définition du système d'information souffre de quelques défauts : les demandes des maîtrises d'ouvrage sont souvent inflationnistes et versatiles et les exigences de l'organisation interne pèsent sur la définition fonctionnelle au détriment des considérations économiques. Même si une partie des référentiels est en bon ordre (référentiel de l'organisation, des métiers), il reste à définir un référentiel des services internes et externes et à rassembler les tables de références dispersées dans les applications. Les discussions stratégiques sur le système d'information sont parfois confuses, les débats portant trop sur les questions techniques. L'économie de l'informatique est scandée par l'annualité budgétaire.

* *

Cette situation (migration massive, progrès en cours, méthodes à améliorer) a incité Deutroisix à lancer une démarche d'urbanisme (Jean [91]). Cette démarche vise à fournir une « portée de phares » de trois ans pour

1. Il s'agit d'un cas réel ; j'ai modifié le texte pour préserver l'anonymat de l'entreprise.

mettre le budget annuel en perspective et anticiper à grosses mailles les évolutions futures. Il a été décidé de nommer l'exercice « schéma d'évolution du système d'information » (SESI) plutôt que « schéma directeur » pour éviter la connotation autoritaire qui s'attache au terme « directeur » et pour rendre compte du fait qu'il s'agit non d'un plan quinquennal, mais d'un exercice à renouveler tous les ans pour tenir à jour la zone éclairée par la « portée de phares ».

Les travaux, réalisés avec l'aide de la société *S*, ont commencé en novembre 2000, le livrable final devant être fourni à la fin de 2001. Ils ont comporté quatre phases dont les dénominations parlent d'elles-mêmes : « état des lieux », « objectifs prioritaires », « architecture cible », « plan d'action ». Chacune de ces phases a donné lieu à des rapports qui ont été validés par les experts et les dirigeants des métiers de l'entreprise selon une logistique compliquée (et coûteuse en ressources internes) de réunions, circulation des documents, recueil des remarques, prises en compte, validation etc. Le livrable final (« plan d'action ») doit contenir un programme d'actions caractérisé par son contenu et ses échéances ainsi que des évaluations de coût, le tout sur les trois prochaines années.

La démarche a été animée par la Mission Architecture de la DSI (maître d'œuvre) et la MOAD auprès du directeur général (maître d'ouvrage), les travaux relatifs à chaque métier de l'entreprise se déroulant sous la conduite du MOAD du métier. Un comité de pilotage a réuni chaque mois ces intervenants et les représentants de la société *S*, ainsi que le DSI et le directeur des études à la DSI.

* *

Le schéma d'urbanisme a permis de mettre en circulation une représentation d'ensemble du système d'information et de son évolution selon des cartographies qui le détaillent par zooms successifs, de la vue d'ensemble jusqu'au plan intérieur de chaque domaine, les évolutions prévisibles étant représentées sous la forme d'une évolution de la cartographie. Les schémas sont accompagnés de commentaires qui en précisent le contenu et expliquent les choix de priorités sous-jacents.

Cet exercice a fourni aux spécialistes de chaque domaine une vue d'ensemble qui permet de percevoir la solidarité entre les domaines et l'importance des flux transverses : les questions d'architecture que posent la gestion des données de référence et les règles de mise à jour des bases de données, ont acquis ainsi l'évidence palpable qui leur faisait défaut. De même, les apports de l'informatique de communication aux processus de production sont plus aisément visualisés. Enfin les contraintes de synchronisme apparaissent mieux. La présentation d'ensemble du système d'information a procédé, de façon systématique, en partant du domaine couvert par chaque métier, en identifiant les processus de production à l'œuvre dans chaque domaine, puis les activités concourant à ces processus, les composants mobilisés par ces activités, enfin les données rassemblées par ces composants (figure 13.1).

On est proche ici du langage UML que Deutroisix utilise désormais pour modéliser ses processus. On en reste toutefois aux niveaux les plus agrégés du langage : ce sont ceux qui conviennent à l'urbanisme.

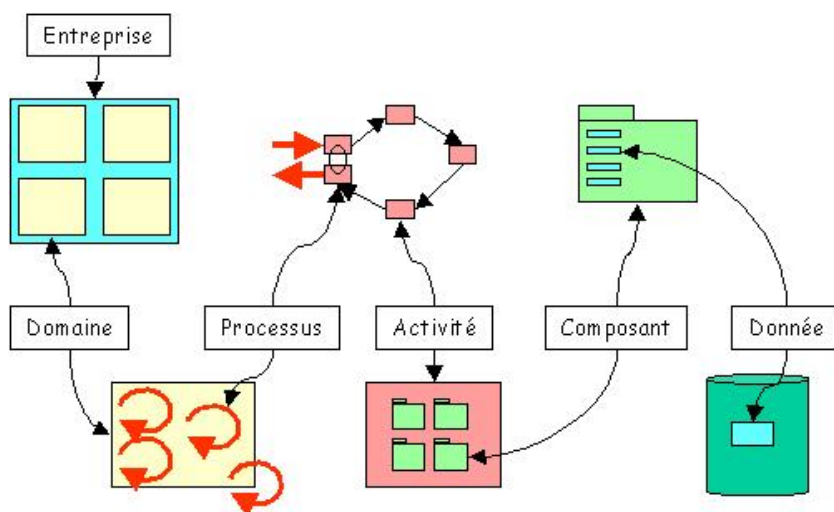


FIG. 13.1 – Étapes de l'urbanisation

L'appropriation du schéma d'évolution par l'entreprise reste cependant problématique. Matériellement, les livrables se présentent sous la forme de classeurs dont la lecture est fastidieuse : ce ne sont pas des documents destinés à la communication ni moins encore à la vulgarisation, mais des textes qui enregistrent le résultat d'un travail professionnel et technique. La pertinence des analyses, l'exactitude des descriptions, la qualité des recommandations ne suffisent pas pour établir un texte dont l'intérêt éveillera l'attention du lecteur et alimentera son intuition. C'est plutôt, il faut le dire, la somnolence qui vous gagne quand vous parcourez ces énumérations et examinez ces graphiques.

Deutrois est donc confrontée à plusieurs risques : au pire, le schéma d'évolution jaunira sur des étagères et nourrira la critique rongeuse des souris. Mais il pourrait encore servir de référence documentaire lors des discussions annuelles, être tenu à jour, sans pour autant être véritablement assimilé par l'entreprise, le cercle de ses lecteurs se limitant à quelques professionnels de la maîtrise d'ouvrage ou de l'informatique.

Il reste donc à trouver la méthode qui permettra d'élargir le cercle des lecteurs et utilisateurs du SESI. Sans ambitionner de toucher tout le personnel de l'entreprise on peut viser à faire partager les représentations et le vocabulaire du SESI par les dirigeants et leurs collaborateurs immédiats (directeur général, DGA, DSI, maîtres d'ouvrage délégués, maîtres d'ouvrages opérationnels, responsables de domaine et chefs de projets de la maîtrise d'œuvre, directeurs régionaux et leurs collaborateurs), soit en tout une petite centaine de personnes auxquelles on peut ajouter le cercle des experts métier et des experts de l'informatique, soit encore quelques centaines de personnes. Il faudrait pour cela pouvoir *médialiser* le SESI, le mettre sur un support de communication qui aide l'utilisateur non seulement à comprendre, mais à *réaliser* ce qu'impliquent les choix effectués - et donc à en

tirer les conséquences. On ambitionne d'aider les utilisateurs à surmonter ainsi l'obstacle que constitue l'apparente abstraction du système d'information et à en percevoir les dimensions pratiques².

* *

Outre la mise en ordre du système d'information, l'exercice d'urbanisme a fait apparaître des enjeux concernant le positionnement de l'entreprise, que ce soit en creux ou en relief :

1) Faut-il ou non diversifier l'offre de services en allant au delà de la plate-forme que constitue le « cœur de métier » ; faut-il ou non enrichir l'intermédiation du marché M en prenant en compte, outre les pôles « offre » et « demande » sur ce marché, d'autres pôles constitués par des offres annexes ? Si la réponse est « oui », cela compliquera le schéma de l'intermédiation ; par ailleurs la disponibilité de l'information sur les offres annexes est problématique.

2) Faut-il articuler le système d'information avec les services rendus sur l'Intranet ? Si la réponse est « oui », cela implique une reconception des processus de production et une modification des fonctions d'encadrement.

3) Faut-il intégrer les services rendus sur la Toile avec les services rendus en face-à-face avec les clients ? La réponse est sans doute « oui », mais ce ne sera pas simple.

4) Les partenaires de l'entreprise s'équipant de système d'information de plus en plus perfectionnés, les partenariats devront s'enrichir d'une prise en compte plus complète de leur identité professionnelle et institutionnelle.

Au total, l'exercice d'urbanisme aura permis d'*élucider* l'évolution du système d'information de l'entreprise et d'éclairer la stratégie de celle-ci. Si on réussit son appropriation par l'entreprise, on aura modifié la façon dont elle perçoit sa relation avec son système d'information et fait progresser la prise de conscience des enjeux professionnels et stratégiques.

13.2 Approche par les processus

On commence à parler en termes de *processus* quand, après avoir répondu à la question « que produit l'entreprise ? », on pose la question « comment produit-elle ? ».

L'approche de l'entreprise par les processus n'a rien de nouveau : elle colle au bon sens, qui n'est pas chose récente ; mais mettre l'accent sur le processus en tant que tel, ce n'est pas seulement dire que l'on va s'occuper de « la façon dont on produit » ; c'est dire aussi que l'on va la documenter, l'explicitier, la *modéliser*. La nouveauté, ce n'est pas de parler des processus - puisqu'il existe des processus depuis que l'être humain travaille pour produire - c'est de se donner pour but de les *élucider*.

2. Il est regrettable que la DSI de Deutroisix n'ait pas su percevoir les services que peut rendre un produit comme OnMap (www.onmap.fr), et qu'elle ait ainsi refusé la médiatisation du SESI.

Élucider un processus, c'est porter sa description à un niveau de *clarté* et de *simplicité* qui l'illumine puis rendre cette description *transparente* pour faciliter son partage entre les acteurs qu'il concerne. Faire passer l'organisation des tâches de l'*implicite* à l'*explicite* a des conséquences : ce n'est donc pas neutre.

On dit parfois qu'il faut « optimiser les processus ». L'ambition d'*optimiser* tend la volonté vers la recherche de la perfection et implique que l'on soit capable de déterminer l'optimum *a priori*. Il nous semble préférable de suivre la démarche plus modeste de l'élucidation.

Un de ses premiers résultats est de rendre visibles les éventuels défauts du processus :

1) Si le traitement du courrier comporte une phase pendant laquelle la lettre arrivée est posée sur une pile puis la pile traitée de haut en bas (comme cela peut se produire sur le bureau de chacun), le courrier sera traité selon le mode LIFO (*last in, first out*) qui introduit des délais aléatoires et suscite des non réponses une fois le délai décent dépassé ;

2) si l'enchaînement des tâches ne permet pas le suivi d'une affaire (on ne peut pas savoir où est un dossier en cours de traitement, on ne sait pas quelles opérations ont été faites sur ce dossier et on ne peut pas consulter les avis donnés avant que l'ensemble du circuit n'ait été parcouru), on devra se livrer sur chaque dossier terminé à une vérification lourde, on risque d'avoir à relancer des démarches parce telle étape intermédiaire aura été mal réalisée ;

3) si l'enchaînement des tâches ne « boucle » pas (c'est-à-dire si l'on ne contrôle pas son délai de traitement), on risque que le dossier se « perde dans les sables », passe de mains en mains pour finir dans une discrète corbeille à papier ;

4) si les « livrables » intermédiaires (ce que doit fournir chacun des acteurs qui contribue au processus) sont définis de façon ambiguë, des allers et retours et récriminations entre acteurs successifs sont inévitables ;

5) si les données sont saisies sans que l'on dispose de moyens de vérification sur le poste de travail, des erreurs seront introduites dans les fichiers ; il faudra les détecter en batch et les corriger péniblement ;

6) s'il faut une ressaisie manuelle pour recopier dans une application certaines données résultant d'une autre, elles provoqueront des erreurs (le taux d'erreur est de quelques fractions de pour mille) et introduiront des délais aléatoires dans les mises à jour ;

7) si une liste de diffusion n'est pas mise à jour sans délai en fonction des nominations et changements d'affectation, les circuits des documents et des informations seront mal ajustés.

8) si une personne ne consulte jamais sa boîte aux lettres, elle sera inopérante dans tout circuit de décision impliquant l'usage de la messagerie ;

9) si une documentation technique est diffusée en mode papier, la prise en compte des corrections apportées par ses versions successives supposera un travail de classement pénible de la part de l'utilisateur : souvent elle ne sera pas faite ;

10) si les données de référence sont stockées dans plusieurs endroits différents, il faudra les mettre à jour à la main simultanément lors de tout changement du contexte, ce qui entraîne des risques d'oublis suscitant des incohérences dans le système d'information, etc.

L'élucidation des processus ne comporte donc pas seulement la phase descriptive pendant laquelle on note ce qui se passe dans des diagrammes avec cases, titres, flèches entre les cases et commentaires ; c'est aussi une phase *normative*, mais très naturelle, car elle fait apparaître des défauts qui sautent aux yeux et que les participants aux travaux trouveront tout simple de corriger.

« Faire apparaître », « trouver tout simple », cela ne va pas de soi : pour que cela marche il faut un animateur habile qui rendra les choses perceptibles en éveillant l'intuition des participants.

La collecte de l'information sur les processus, la validation de leur élucidation, la discussion des résultats, supposent des contacts avec des experts du métier puis une animation plus large touchant l'ensemble des praticiens concernés. L'expérience montre que les gens de métier participent avec enthousiasme à ces démarches. Le mot *enthousiasme* n'est pas trop fort : l'élucidation clarifie des questions qu'ils se posaient confusément et qui les mettaient mal à l'aise ; elle leur permet de supprimer des dysfonctionnements irritants, ou de comprendre la rationalité sous-jacente à ce qu'ils prenaient pour un dysfonctionnement.

Si l'on veut capitaliser le progrès accompli lors de l'élucidation *il faut articuler une transformation du système d'information à l'élucidation du processus*. C'est ce que certains consultants résumant par la règle « pas de processus sans workflow », *No Process Without Workflow*.

Il y a là une difficulté pratique. Les consultants spécialisés dans l'élucidation des processus (on dit « l'analyse des processus ») sont souvent des organisateurs, non des informaticiens, et il existe au sein des grands cabinets une méfiance réciproque entre consultants en organisation et consultants en informatique. À cette difficulté correspond un piège : d'excellents travaux peuvent être réalisés sur les processus sans que l'on se soucie de leur articulation avec le système d'information. Trop souvent ces travaux se concentreront sur des améliorations de détail et de court terme, utiles certes mais d'une utilité limitée car ils n'envisagent pas la totalité du processus mais seulement une partie ; par ailleurs, il sera impossible, sur la base de ces travaux, de disposer des outils de contrôle automatique qui seuls permettraient de vérifier leur application.

* *

Un processus, c'est la succession des activités des acteurs qui contribuent à l'élaboration d'un produit. Le « workflow », c'est l'outil informatique qui rend *visible* cette succession et permet le contrôle de ses délais, de sa qualité et de son efficacité. Lorsqu'un processus est équipé d'un workflow, il est possible de :

- savoir à tout moment où en est la procédure appliquée à un dossier, consulter les avis qui ont été donnés, relancer la procédure si elle s'enlise,

- contrôler les délais de bouclage sur l'ensemble du processus comme sur la production des livrables intermédiaires,
- rerouter automatiquement un dossier si l'un des acteurs est absent ou empêché, ou encore s'il met trop de temps à le traiter,
- produire automatiquement des indicateurs de délais et de volume permettant à l'animateur de contrôler la qualité du processus et de redéfinir l'allocation des ressources si des goulets d'étranglement apparaissent.

Le workflow, avec son formalisme et ses outils, concrétise l'élucidation du processus et permet de la capitaliser, c'est-à-dire de la garder vivante en mémoire, de la mettre à jour et de la communiquer.

Comment se présente une entreprise dont les processus ont été élucidés et équipés de workflows permettant un contrôle de leur qualité (ainsi que les évolutions liées à celles des missions de l'entreprise)?

1) d'une part, et c'est ce qui frappe le plus, les personnes qui travaillent dans cette entreprise trouvent leur travail *simple*, logique, efficace. L'organisation de la succession des tâches leur est connue ainsi que la démarche à suivre en cas d'incident. Elles savent ce qu'elles ont à faire et ce que doivent faire les personnes avec qui elles coopèrent. Elles disposent ainsi d'un espace de responsabilité dans lequel elles peuvent exercer leur jugement et leur esprit d'initiative ;

2) le nombre des niveaux hiérarchiques a diminué : la transparence, l'automatisation des indicateurs rendent en effet inutiles les « petits chefs » dont le rôle était de transmettre aux exécutants les consignes de la direction, puis de faire des rapports sur l'exécution des tâches ;

3) les structures ainsi organisées sont « qualifiantes » : l'agent accroît ses compétences en travaillant ;

4) leur pilotage est facilité par la transparence du processus. Cette transparence favorise la qualité en faisant clairement apparaître les performances de chaque entité ;

5) les informations accumulées à l'occasion du déroulement des processus constituent une source éditoriale qui peut être exploitée pour alimenter les systèmes d'information décisionnels³.

* *

Pour décrire l'activité d'un utilisateur, il suffit d'indiquer les données que celui-ci consulte, celles qu'il saisit, les traitements qu'il lance, ainsi que l'ordre (éventuellement souple) dans lequel il réalise ces opérations. Chaque utilisateur va consulter ou saisir quelques données, déclencher un nombre limité de traitements: ceci conduit naturellement vers la programmation orientée par objets.

Le langage UML (Fowler [181]), qui fédère les langages de modélisation en matière d'approche objet, fournit les documents nécessaires pour décrire les activités (« use cases » selon le vocabulaire de Jacobson), les classes

3. L'étude du cas « Infotel » illustre les conséquences pratiques de cette démarche ainsi que les difficultés que l'on rencontre dans son déroulement (voir page 375).

(« diagrammes de classes ») et la succession des opérations (« diagrammes de séquences »). On construit ainsi un « modèle complet » qui, établi par un maître d'ouvrage et communiqué au maître d'œuvre informatique, indique à ce dernier ce qui doit apparaître sur les écrans des utilisateurs, les actions que ceux-ci vont réaliser, ainsi que les compteurs utilisés à des fins de contrôle.

Ce *modèle* indique sans ambiguïté ce que l'utilisateur veut faire et aide à découper le développement en petits modules, les classes, clairement reliées chacune à une finalité pratique (c'est pour cela que l'on parle d'« objets métier »).

L'analyse des activités fait apparaître que les mêmes classes sont utiles à plusieurs acteurs ou que l'on peut satisfaire leurs besoins en construisant des classes de forme identique ou analogue (héritage, polymorphisme). Elle fait également apparaître que les mêmes classes sont souvent articulées au sein de « composants » qui les regroupent, ce qui permet de fédérer leurs interfaces. Ces analogies et regroupements dégagent des êtres sémantiques nouveaux, concrétisant des concepts inédits. Les langages à objets exploitent ces possibilités, ce qui allège le développement au bénéfice de la conception. Ils visent à réduire ainsi les coûts de maintenance et à faciliter l'évolution du système d'information.

La mise en œuvre du modèle par l'informatique suppose que les développements soient réalisés en utilisant un des langages à objets (Java, C++). Une articulation intelligente entre outils de développement et langage de modélisation permet de maîtriser la documentation des versions successives, les mises à jour à introduire lors des évolutions, et facilite l'évolutivité du logiciel selon les versions successives du modèle métier.

Les échanges de messages entre objets ou avec les bases de données sont réalisés par un *middleware* qui traite les problèmes d'adressage, de transcodage, de synchronisme, ainsi que les questions de « persistance » pendant la durée de la transaction et de « concurrence » lorsque deux utilisateurs travaillent en même temps sur un même composant. Ceci suppose que l'informatique soit capable de dominer l'articulation avec les anciennes applications dont l'adaptation aux langages à objets prendra quelques années. Les techniques nécessaires pour assurer un service de qualité avec des langages hétérogènes nécessitent une compétence élevée en informatique.

Cette évolution rend nécessaire l'acquisition de compétences nouvelles par les informaticiens (apprentissage des langages Java et C++, maîtrise du « middleware », de la programmation des interfaces, de l'articulation de la gestion de configuration avec les modèles métiers : elle offre ainsi à la profession informatique un terrain d'expansion, et renouvelle les conditions de son dialogue avec les utilisateurs.

13.3 Modélisation

Un « modèle », c'est une description d'un être réel conçue de telle sorte qu'il soit possible de simuler mentalement le fonctionnement de cet être. Tout modèle comporte à la fois des *concepts* qui permettent de décrire l'être

en question et des *relations causales* entre concepts. Un modèle possède donc tous les attributs d'une théorie mais, dans le cas du système d'information, il s'agit d'une *théorie orientée vers l'action*, d'une théorie à finalité pratique.

Chacun modélise tout le temps car c'est ainsi que notre cerveau se représente le monde et assiste notre action ; mais cette activité, aussi naturelle que la respiration ou la digestion, reste presque toujours *implicite* : comme la pensée est une activité naturelle peu de personnes réfléchissent à son fonctionnement. Seules la science et l'entreprise exigent des modèles *explicites*, documentés, comportant la définition formelle des concepts et hypothèses de causalité. L'explicitation du modèle est en effet nécessaire au partage d'une même pensée entre plusieurs personnes, ou à l'intérieur d'une institution, en vue d'une action coordonnée : elle a donc, tout comme le modèle, une finalité pratique. Toutefois un modèle explicite est difficile à comprendre et à communiquer : malgré sa finalité pratique, il semble « abstrait », son schématisation étant aussi opaque pour le non-initié que (par exemple) le plan d'un appareil électronique alors que ce plan est pourtant nécessaire à la construction de l'appareil.

* *

La maîtrise d'ouvrage est à la charnière entre le stratégique et l'opérationnel : elle doit éclairer le stratège dans sa fonction d'arbitrage et d'orientation en lui apportant des informations sur la pratique des métiers de l'entreprise ainsi que le résultat d'une veille sur l'état de l'art des systèmes d'information. Elle facilite ainsi l'incarnation opérationnelle des priorités de l'entreprise.

Comme tout autre actif de l'entreprise, le système d'information doit être d'abord conçu et mis en place : c'est la phase des « projets » qui sont un investissement. Une fois en place, le système d'information doit être géré. Le rôle de la maîtrise d'ouvrage dans la phase d'investissement se résume par le mot « modélisation », et dans la phase de gestion par le mot « animation ».

Le succès ou l'échec d'un développement informatique se jouent, avant même que l'écriture du code proprement dit n'ait débuté, lors de la phase de modélisation : l'entreprise y définit ses concepts, ses processus, les contours de son automatisation, la façon dont elle articule l'automate et l'organisation du travail humain.

Une fois l'automate en place il faut que l'entreprise s'approprie le système d'information : les choix faits lors de la phase de modélisation en ce qui concerne les concepts, les processus, l'articulation entre l'être humain organisé et l'automate, doivent être compris par les acteurs - ou mieux, ils doivent être « réalisés », les acteurs devant comprendre intuitivement à la fois leur *réalisme* et leur *réalité* - le *réalisme* du modèle, s'il parvient à atteindre l'efficacité au moindre coût ; et, par delà son apparence théorique, la *réalité* de ce qu'il schématise et qui est l'entreprise même. Il en résulte, outre la diffusion des « bonnes pratiques », une *élucidation* de l'entreprise : chacun voit clairement ce qu'il a à faire, l'articulation entre sa propre tâche et celle des autres, enfin la finalité du processus auquel il contribue et la place de celui-ci dans la stratégie de l'entreprise.

Dans l'accomplissement de ces missions de modélisation et d'animation, le maître d'œuvre (ou, autrement dit, la direction informatique de l'entreprise) est le partenaire naturel de la maîtrise d'ouvrage. Le maître d'œuvre est responsable de la plate-forme technique, physique du système d'information ; la définition et la gestion de cette plate-forme, très complexe, est parallèle à celle du système d'information auquel elle impose ses contraintes. Le maître d'ouvrage doit être, pour le maître d'œuvre, un « client compétent », qui sache exprimer clairement ses besoins, puis comprendre les contraintes propres à la plate-forme technique et en tenir compte.

La plate-forme utilise des ressources physiques (processeurs, mémoires, réseaux) sur lesquels elle met en œuvre des logiciels (systèmes d'exploitation, programmes applicatifs, protocoles de communication). Son fonctionnement suppose, après l'investissement initial et son déploiement, un travail continu d'exploitation (opération, supervision, maintenance, aide aux utilisateurs).

Les utilisateurs accèdent à la plate-forme à travers l'interface que fournit le poste de travail en réseau. Sur cette plate-forme, ils mettent en œuvre des automates qui assistent le processus de production, ainsi que des outils de communication interpersonnelle (messagerie, Intranet). Les services ainsi fournis s'appuient sur le socle sémantique de l'entreprise, le *référentiel*, qui définit les identifiants, les nomenclatures et le plan d'urbanisme lui-même.

Un même système d'information sera perçu par chacun des deux partenaires à travers des vues différentes, l'un se concentrant sur les conditions physiques de son fonctionnement, l'autre considérant les services qu'il utilise. L'interface homme-machine étant la charnière entre ces deux vues, la *qualité du poste de travail* est l'un des enjeux principaux du système d'information.

13.3.1 « Modéliser » l'entreprise

Le modèle d'une entreprise doit pouvoir être partagé par les divers acteurs que comporte cette institution : il doit être *explicite*. Modéliser l'entreprise, ce sera donc d'abord documenter les tâches qui y sont réalisées tant par l'être humain que par l'automate (le modèle, notons-le, ne doit pas se limiter à la description des tâches automatisées car seule une partie des tâches qu'assure l'entreprise est réalisée par l'automate).

Ce travail documentaire suppose que l'on clarifie le vocabulaire, que l'on explicite les procédures et les contraintes auxquelles elles doivent obéir ainsi que les rôles et responsabilités des acteurs. Il comporte donc une mise en ordre du langage (suppression des homonymes et synonymes, mise en lumière des concepts fédérateurs) comme de l'organisation (clarification des missions des entités, de l'attribution et de l'enchaînement des tâches).

* *

Le socle sémantique du modèle est le *référentiel*, qui enregistre le vocabulaire de l'entreprise (identifiants, nomenclatures) et ses règles de gestion.

L'*urbanisme* est une modélisation globale de l'entreprise : il identifie ses grands domaines de production et, à l'intérieur de ces domaines, les proces-

sus et leurs relations mutuelles (certains processus sont « transverses » à divers domaines).

Enfin, *chacun des processus de production* doit lui-même être modélisé pour documenter l'enchaînement des activités humaines et des opérations informatiques, ainsi que le cycle de vie et les règles de gestion des dossiers (ou « objets » en informatique) que le processus manipule. Le modèle d'un processus doit spécifier les éléments que comporte la figure 13.2.

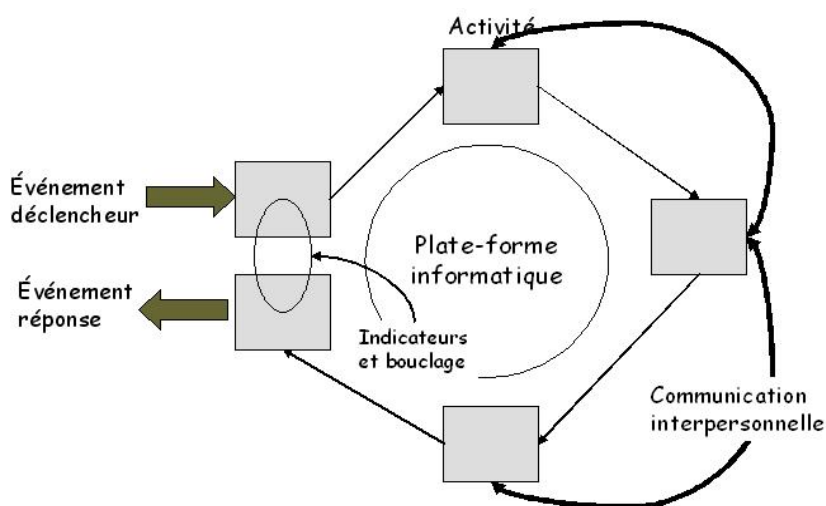


FIG. 13.2 – Modélisation d'un processus

- l'événement extérieur au processus et qui le déclenche (réception d'une commande, d'une lettre de réclamation etc.) ;
- l'enchaînement des activités qui constituent le processus, en précisant les pré-et post-conditions du passage d'une activité à la suivante ;
- les relations entre chacune des activités et la plate-forme du système d'information (données saisies, données consultées, traitements lancés) : la figure 13.2 fait apparaître que les activités « font la ronde » autour de la plate-forme, qui assure la cohérence des données pendant le déroulement du processus ;
- les éventuelles communications interpersonnelles entre acteurs du processus (messages, échanges de documents) ;
- les indicateurs de volume (charge des ressources, productivité) et de qualité (respect des délais, satisfaction du client) fournis au responsable du processus et qui servent à maîtriser la qualité de celui-ci notamment à l'occasion du « bouclage » du processus (contrôle de bonne fin).

La cellule élémentaire du processus, c'est ce que nous avons nommé « l'activité » ; il est utile de la regarder de près (figure 13.3).

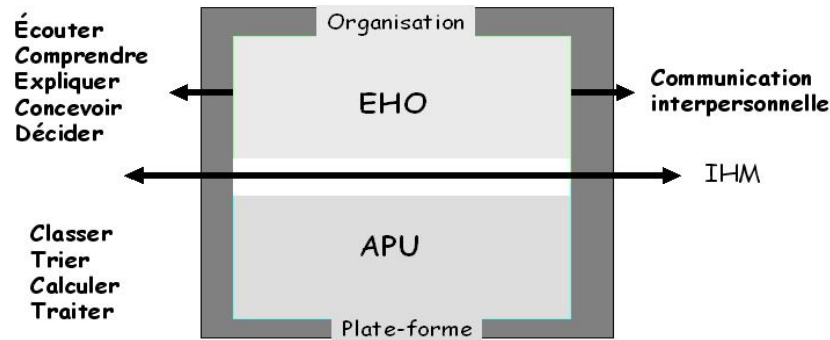


FIG. 13.3 – Anatomie de l'activité élémentaire

Chargée de réaliser une tâche au sein du processus, l'activité met en relation l'être humain organisé (EHO) et l'automate programmable doué d'ubiquité (APU) à travers l'interface homme-machine (IHM) qui est comme une synapse entre les deux parties du système. Chacun des deux acteurs est complémentaire de l'autre : l'APU classe, trie, calcule et traite ; l'EHO écoute, comprend, explique, conçoit et décide. Enfin, alors que la centralité du système d'information au cœur du processus garantit la cohérence des données dans l'APU, la communication interpersonnelle des EHO favorise la cohérence du discernement et de la décision.

* *

La modélisation obéit à deux finalités, l'une technique, l'autre intellectuelle.

Au plan technique, le modèle précise la conception du système d'information : son élaboration est la première étape de sa mise en place, tout comme celle du plan d'un immeuble est la première étape de sa construction. Il doit aussi préciser les conditions de fonctionnement du système d'information.

Au plan intellectuel, la modélisation permet à l'entreprise de partager une représentation de son propre fonctionnement : les objets, concepts, processus et référentiels de l'entreprise sont élucidés. Cette clarté permet de compenser l'obstacle à la communication et à la compréhension que représente le cloisonnement en spécialités et, au plan psychologique, l'autisme professionnel qui est si fréquent chez les spécialistes. Un modèle bien fait et convenablement approprié par l'entreprise, permet à chacun de se représenter le cours du processus pour lequel il travaille, son propre rôle dans le processus ainsi que les rôles des autres acteurs, enfin les relations entre les divers processus de l'entreprise.

Les deux finalités sont aussi importantes l'une que l'autre, ou plus précisément elles se situent sur des plans différents où elles sont toutes deux cruciales. On a trop tendance à négliger la finalité intellectuelle parce que l'on ne voit dans le modèle qu'une étape d'un projet technique.

Beaucoup d'entreprises ne modélisent pas car *elles croient inutile de comprendre comment elles font ce qu'elles savent faire* (il est vrai que per-

sonne ne sait comment son corps respire ou digère et que cela n'empêche pas de vivre). Lorsqu'un salarié qui arrive dans une telle entreprise est mis au travail⁴, il lui est demandé d'adhérer à une organisation locale sans qu'il puisse en prendre la vue d'ensemble. Il se formera par imitation des anciens : le savoir est « dans les murs ». L'architecture qui résulte des choix faits dans le passé par des organisateurs est considérée comme un état de la nature. Lorsqu'on présentera un modèle à ce salarié, il le trouvera déroutant car le modèle relativise des conventions et règles d'organisation qu'il a l'habitude de considérer comme des absolus.

Certaines entreprises doivent cependant modéliser : celles qui sont placées dans un contexte évolutif (concurrence, innovation technique, réglementation) et doivent donc être *agiles* ; ou encore celles qui partagent avec d'autres entreprises la production d'un produit (partenariats) et doivent donc assurer l'*interopérabilité* des processus.

Or aujourd'hui la plupart des entreprises se trouvent ou se trouveront bientôt dans l'un ou l'autre de ces deux cas : l'évolutivité et les partenariats sont des contraintes de l'économie actuelle. Ainsi, alors qu'autrefois les entreprises pouvaient à peu près bien marcher en inculquant à leurs salariés des habitudes professionnelles, il importe maintenant qu'elles élucident leurs processus afin que leurs salariés puissent se les approprier⁵.

13.3.2 Démarche de modélisation

Un modèle exprimé doit décrire complètement le contenu fonctionnel d'un programme informatique⁶. Le langage UML, bien adapté à la technologie objet, définit sur ce modèle des vues graphiques, ou diagrammes, qui doivent être complétées par une documentation en langage naturel.

La réalisation du modèle doit progresser par étapes caractérisées chacune par la confection d'un diagramme, la démarche procédant par enrichissement progressif selon un ordre descendant (*top down*). Il ne convient pas de bousculer l'ordre des étapes : il ne faut pas se lancer dans la modélisation proprement dite sans disposer de l'expression de besoin, ni documenter les cas d'utilisation avant d'avoir produit le diagramme d'activité etc. Chaque étape de la démarche doit avoir été réalisée entièrement avant que l'on ne passe à la suivante : l'une des erreurs les plus courantes en modélisation est de s'engouffrer de façon précoce dans des travaux détaillés qu'il faudra détricoter par la suite. Il sera nécessaire d'associer plusieurs techniques informelles et formelles pour saisir les diverses facettes d'un problème sans

4. Il se peut qu'alors l'entreprise « forme » le salarié. Mais la mise en scène d'une formation n'est pas la même que celle du travail ; il y a, entre la formation dans une salle de cours et l'utilisation d'un modèle sur le lieu de travail, le même écart qu'entre les verbes « comprendre » et « réaliser », ce dernier ajoutant à la compréhension l'intuition du caractère réel de ce que l'on comprend.

5. La recherche de la productivité peut, elle aussi, conduire l'entreprise à modéliser : c'est le cas lorsque l'organisation « gravée dans les murs » n'assure plus la conformité à l'état de l'art du secteur.

6. Je reprends ici les conclusions d'un groupe de travail du club des maîtres d'ouvrage www.clubmoaasso.fr et les travaux de Philippe Desfray, de la société Softeam.

le dénaturer, puis pour le détailler dans un modèle central que l'on pourra modifier jusqu'à l'implémentation. L'utilisation conjointe de ces diverses techniques permet de s'adresser à des interlocuteurs qui diffèrent tant par la forme de leur intuition que par le niveau de visibilité et de lecture des modèles (figure 13.4).

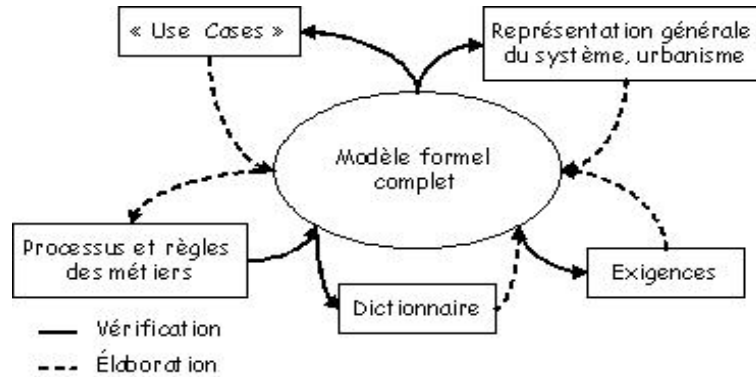


FIG. 13.4 – Les étapes de la démarche

Il se peut que les contraintes que l'on découvre lors de la réalisation de l'étape n de la démarche obligent à réviser les résultats des étapes antérieures. Ainsi, alors que la démarche doit progresser sans que l'on n'anticipe le travail à faire dans les étapes ultérieures, elle ne peut pas exclure les retours en arrière. Il faut d'abord réviser l'étape la plus en amont parmi celles qui sont à revoir, puis revenir vers l'aval jusqu'à l'étape en cours. Un modèle bien fait facilite ces révisions : si la documentation est correcte, si les outils facilitent la cohérence entre les étapes, si le modèle est convenablement modulaire, le travail nécessaire aux révisions reste raisonnable.

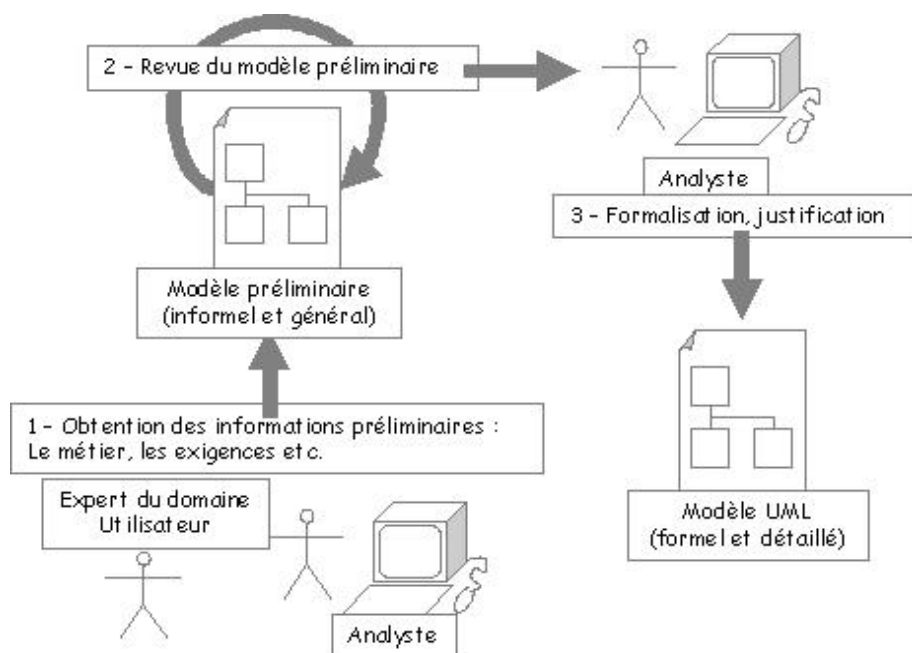
Le processus de réalisation du modèle peut se représenter par une boucle comme dans la figure 13.5.

Chaque étape aboutit à un « livrable » spécifique qui doit être dûment validé, ce qui évite l'effet de tunnel dans la modélisation⁷.

* *

Une fois l'expression des besoins précisée et validée, il convient d'établir en premier le dictionnaire du domaine considéré ; ensuite, une approche systémique en fournit une vue globale. La définition des modèles conceptuels peut accompagner la modélisation des processus métier, enrichie par la prise en compte des règles de gestion. Enfin, les cas d'utilisation détaillent ce que le modèle doit effectuer au sein du système global.

7. L'effet de tunnel doit être évité également dans la réalisation du logiciel : si l'automatisation du processus requiert un travail lourd (et donc long) il sera impératif de définir des « livrables exploitables », produits intermédiaires dont la mise en exploitation entre les mains des utilisateurs fournira des enseignements utiles.

FIG. 13.5 – *Processus de réalisation du modèle**Expression de besoin*

Toute modélisation informatique doit partir d'une expression de besoin écrite, claire et validée par les responsables. Elle servira de référence tout au long de la réalisation du projet (voir page 456).

Lorsque l'expression de besoin part comme il se doit du *produit* que l'entreprise entend élaborer, elle comporte naturellement la description d'un processus de production : toute action se ramène en effet à un processus (plus ou moins complexe et ramifié) selon lequel, partant d'une situation initiale, et par l'application de certaines ressources, on aboutit à un résultat, à un produit.

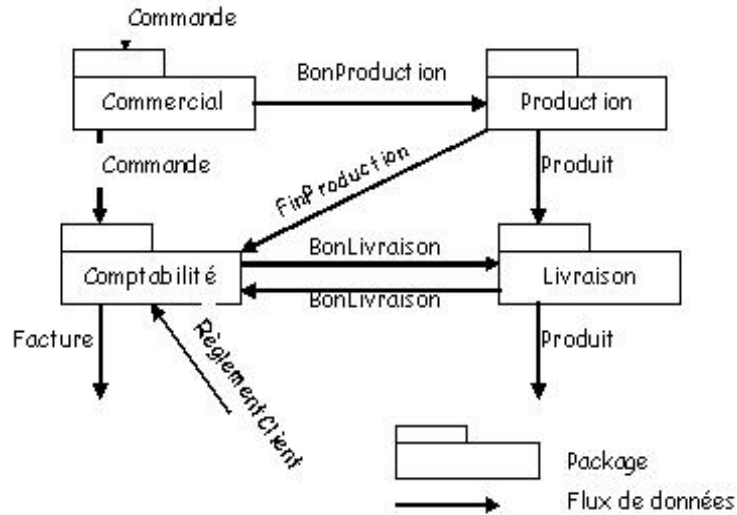
L'expression de besoin doit essentiellement décrire la situation initiale et le résultat, la définition des ressources relevant le plus souvent d'une étape ultérieure. Il faut préciser aussi ce qu'il conviendrait de faire en cas de défaillance : la modélisation du processus doit comporter la description des cas de panne et des solutions à adopter en régime dégradé.

Dictionnaire

Le dictionnaire rassemble les définitions des termes relatifs au système considéré. On doit être tolérant lors du recueil de la terminologie du métier et accepter de noter les homonymes et synonymes qui coexistent dans l'organisation. Cependant la construction du modèle apportera une réduction terminologique en n'associant plus qu'un nom et un seul à une même chose ou à un même concept : l'amélioration du vocabulaire est l'un des apports de la modélisation.

Approche systémique

Il est utile de produire un schéma général, validé par les acteurs, mettant en évidence les structures de l'entreprise impliquées, leurs responsabilités et leur mode de coopération. La notion de « flot d'information » introduite dans UML 2.0 sera ici utile (figure 13.6).

FIG. 13.6 – *Diagramme de flux**Règles métier*

La notion de « contrainte » dans UML permet de modéliser des règles de gestion qui autorisent, provoquent ou empêchent le déroulement d'un processus (exemples: « une direction départementale ne doit pas comporter plus de dix agences », « un client ne peut commander *via* la Toile que s'il a été enregistré au préalable », « un employé marié ne doit être muté qu'en dernier recours » etc.).

Modélisation du processus

Décrire un processus, c'est décrire l'événement qui le déclenche, les étapes (ou activités) par lesquelles il doit passer, les ressources qu'il doit consommer, l'événement final auquel il doit aboutir. Ces informations sont rassemblées et documentées dans le diagramme d'activité qui décrit la succession des activités, les messages qu'elles échangent, les éventuels sous-processus et les livrables intermédiaires que ceux-ci fournissent (figure 13.7).

Le diagramme d'activité est, parmi les diverses vues sur le modèle, la plus lisible et la plus facilement compréhensible pour les personnes qui ne participent pas à son élaboration. Il joue un rôle important lors de sa validation.

Cas d'utilisation

L'étape suivante consiste à décrire les cas d'utilisation, chaque activité en comportant un ou plusieurs. Un cas d'utilisation regroupe des opérations

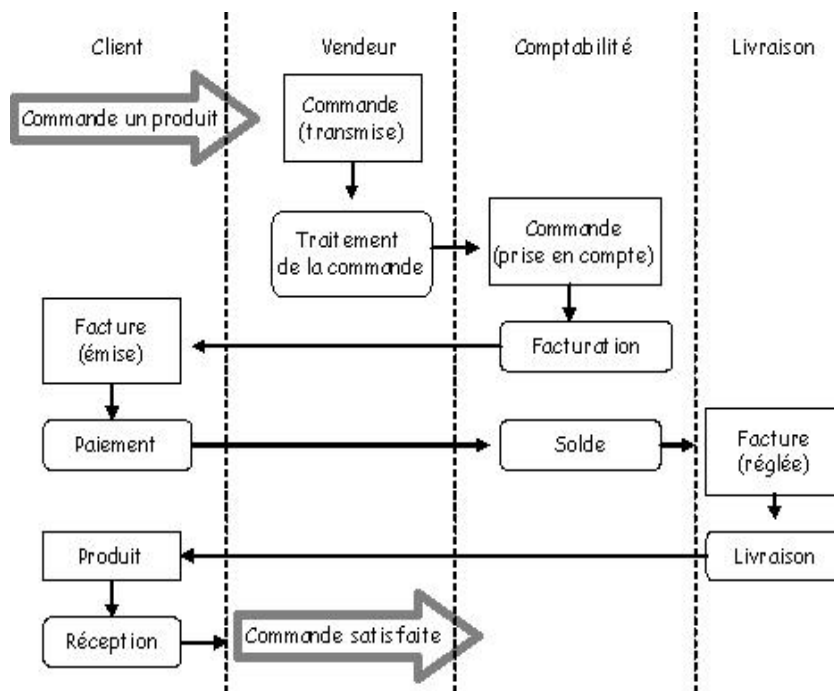


FIG. 13.7 – Diagramme d'activité

que l'acteur exécute et qui forment un ensemble cohérent : recevoir des messages, consulter des données ou du texte, saisir des données ou du texte, lancer des traitements, envoyer des messages. On a défini le cas d'utilisation lorsque (1) on l'a nommé et désigné par sa finalité au sein de l'activité, (2) on a décrit son contenu en définissant les données consultées, saisies ou traitées, la nature des traitements, les messages échangés, (3) on a identifié les composants applicatifs qu'il met en œuvre au sein du système informatique (figure 13.8).

Il arrive que des cas d'utilisation divers comportent des éléments semblables, ou qu'ils soient des cas particuliers de cas d'utilisation plus généraux : on peut alors définir une hiérarchie de cas d'utilisation qui, par abstraction, simplifie le modèle : c'est le diagramme des cas d'utilisation.

Pour valider un cas d'utilisation, on présente aux utilisateurs futurs une succession d'écrans simulant l'exécution du processus.

« *Modèle métier* » et « *modèle complet* »

La maîtrise d'ouvrage est responsable du modèle métier, qui n'est rien d'autre qu'une expression de besoin précise et ne considère les contraintes techniques que de façon approximative. Pascal Roques et Franck Vallée [207] ont proposé un schéma en Y pour la modélisation (figure 13.9). La réalisation du modèle métier occupe la branche gauche ; la prise en compte des contraintes techniques occupe la branche droite ; la réalisation du modèle complet est la branche verticale. Le modèle métier ne peut être que provi-

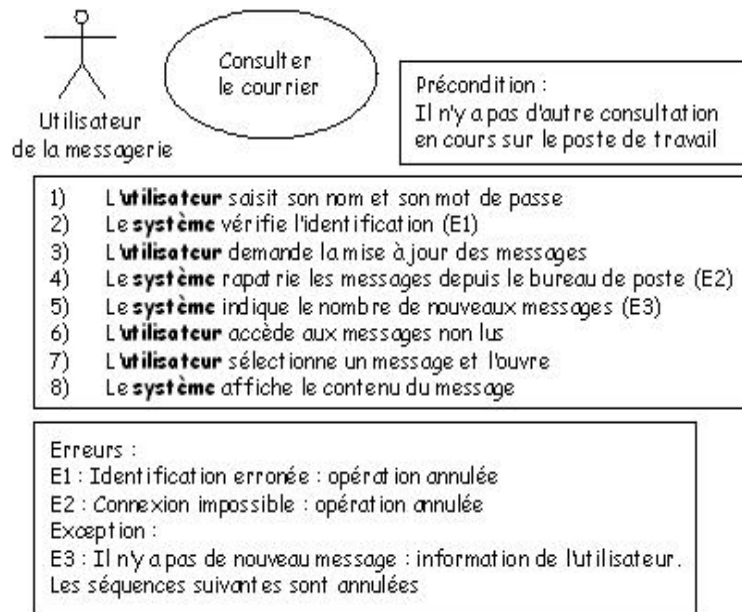


FIG. 13.8 – Cas d'utilisation

soire: il arrive que l'on doive le réviser pour tenir compte des contraintes techniques ou de l'évaluation du coût de réalisation que fait apparaître l'élaboration du modèle complet.

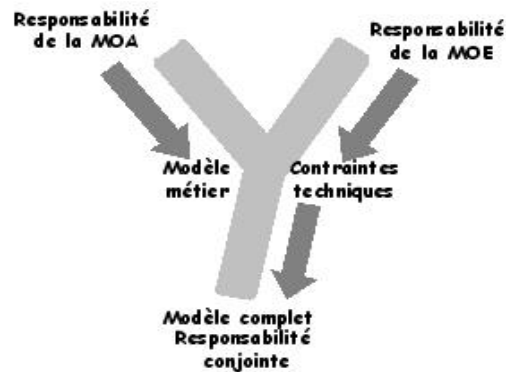


FIG. 13.9 – Rencontre du modèle métier et des contraintes techniques

13.3.3 Quelle est l'utilité du modèle métier ?

Le modèle métier est susceptible d'être modifié par la prise en compte des contraintes techniques. Est-il inutile pour autant ? Supposer le modèle métier inutile, cela reviendrait à dire que l'informatique pourrait travailler à

partir d'une expression de besoin sommaire, avec les ambiguïtés du langage naturel, et qu'il lui reviendrait de modéliser les processus des métiers et de déterminer leurs concepts structurants.

C'est ainsi en effet que les choses se sont passées lorsque l'informatique équipait les grandes applications de support de l'entreprise (paie, comptabilité, gestion des stocks etc.) qui, tout en étant complexes, varient peu d'une entreprise à l'autre. Il ne peut en être de même lorsque l'informatique pénètre les processus de production, lorsqu'elle équipe le cœur de métier sur lequel l'entreprise possède des compétences spécifiques et « pointues ».

La modélisation du processus prendra d'ailleurs des formes diverses selon que celui-ci :

- met en œuvre un automate pur (cas du logiciel embarqué dans une machine),
- associe l'automate à des êtres humains pour mettre en œuvre les procédures internes à l'entreprise (back-office),
- reçoit des messages provenant d'acteurs externes et qu'il faut transcrire dans le système d'information de l'entreprise (cas de la première ligne face à des clients, fournisseurs, partenaires),
- traverse les frontières de l'entreprise pour pénétrer chez des partenaires, fournisseurs ou clients (interopérabilité).

En outre dans chaque cas il faudra tenir compte d'exigences spécifiques de synchronisme, qualité des données et homogénéité des codages. Considérons par exemple les processus qui prennent leur source à la première ligne. La modélisation doit définir conjointement les tâches qui reviennent alors à l'être humain et celles qui reviennent à l'automate. Un des points délicats est de placer raisonnablement la frontière de l'automatisation : cela suppose une connaissance fine des pratiques professionnelles.

La modélisation est par ailleurs pour un métier l'occasion d'une réflexion sur les procédures et d'une révision des processus. Elle permet de mettre à plat des habitudes qui se sont stratifiées dans l'histoire de l'entreprise et dont certaines ont perdu leur raison d'être, de détecter et corriger des défauts (bras mort où se perdent les dossiers, redondances, erreurs d'adressage etc.), de rétablir la qualité des référentiels qui se dégrade toujours au long de la vie de l'entreprise. Il arrive même que la modélisation incite l'entreprise à relancer la réflexion stratégique sur son positionnement, la diversification de son offre et la définition de ses partenariats. Ainsi s'élabore la pertinence de l'expression de besoin, son adéquation aux besoins pratiques du métier.

Ces qualités cruciales, observons-le, dépendent toutes trois de la maîtrise d'ouvrage. Elles sont conditions nécessaires de la qualité du système d'information, mais non conditions suffisantes : il faut encore que la faisabilité technique soit élaborée par la maîtrise d'œuvre, qui doit vérifier si l'automate pourra exécuter les opérations demandées et si l'interopérabilité avec les autres entreprises pourra être assurée. Il en résulte des modifications qui le plus souvent n'excèdent pas 10 à 20 % de la documentation du modèle métier et ne le remettent donc pas en question de fond en comble.

Par la suite, le modèle métier devra s'adapter aux évolutions de l'entreprise. Il est utile de distinguer ici divers niveaux de stabilité. Les choix

techniques ont le niveau de stabilité le plus bas, car ils doivent évoluer rapidement pour assurer le maintien de l'entreprise à l'état de l'art ; le découpage des domaines de légitimité de l'entreprise, que l'on nomme souvent « organisation », un peu plus stable que les choix techniques, est cependant évolutif. Le référentiel des données et des notions que le modèle manipule, étant pour l'essentiel indépendant des choix techniques comme de l'organisation, est la partie la plus stable du modèle et sera le socle sur lequel on peut le bâtir.

13.3.4 Modélisation et évaluation du coût

L'évaluation du coût du projet progresse à mesure que se précise la définition de celui-ci. Considérons les étapes successives dans un cas type, sachant qu'il est susceptible de variantes selon la taille du projet, la maturité de la maîtrise d'ouvrage, la nature interne ou externe de la MOE etc. :

1) au début, l'expression de besoin n'est accompagnée d'aucune évaluation du coût ; on peut tout au plus lui associer une indication qualitative comme « petit projet », « projet moyen », « gros » ou « très gros » ;

2) si l'expression de besoins passe le cap de la première sélection, une « étude préalable⁸ » est lancée. La maîtrise d'œuvre fournit alors, en s'appuyant sur son expérience et sur des « règles de pouce », une esquisse de solution et une première évaluation du coût. La marge d'erreur est de l'ordre de 50 % (si l'on estime le coût à 10 millions d'euros, le coût véritable se situera vraisemblablement dans une fourchette de 5 à 20 millions d'euros : les deux bornes de la fourchette sont dans un rapport 4). Tout imprécise qu'elle soit, cette estimation sert à évaluer la rentabilité du projet (l'estimation des gains que l'on peut en attendre est d'ailleurs, à ce stade, tout aussi imprécise) ;

3) si l'étude préalable est convaincante, l'entreprise lance la première modélisation, opération d'un coût significatif. La réalisation du modèle complet permet de resserrer la fourchette d'évaluation :

- le modèle métier permet de mieux évaluer les gains ainsi que le coût de la maîtrise d'ouvrage (formation, déploiement etc.) ;

- la prise en compte des contraintes techniques permet de mieux évaluer le coût de la réalisation. Le modèle ainsi enrichi, comportant une évaluation plus précise des gains, des coûts et de la rentabilité ainsi qu'un ou deux scénarios d'architecture, constitue le cœur du cahier des charges de la réalisation ;

4) si l'entreprise décide de lancer le projet au vu du modèle complet, la DSI consulte des entreprises. Leurs réponses permettront de préciser encore le scénario d'architecture ainsi que l'évaluation du coût ;

5) il arrive enfin souvent que l'on soit contraint de réviser l'évaluation du coût lors de la réalisation : on ne connaîtra vraiment celui-ci qu'à la fin du projet.

8. Nommée aussi « étude OFR » (« Opportunité, Faisabilité et Risques »).

L'évaluation du coût du projet est donc continue; très floue dans les premières étapes, elle gagne progressivement en précision. Il est bien sûr souhaitable d'utiliser des méthodes aussi rigoureuses que possible, un « modèle de coût » étalonné et condensant l'expérience des informaticiens. Cela ne supprime pas l'incertitude, mais cela peut la réduire et améliorer d'autant la qualité des décisions. Il est d'ailleurs salubre, lorsque les décisions sont prises sur la base d'une évaluation imprécise, que cette imprécision soit considérée comme l'un des risques associés au projet.

13.4 Réalisation des projets

Les projets nouveaux focalisent l'attention des entreprises au détriment de l'utilisation des ressources existantes. Dans les formations aux systèmes d'information, les méthodes de conduite de projet (MCP) occupent une place démesurée.

De fait, les projets qui échouent ou déçoivent sont beaucoup plus nombreux que ce que l'on tolérerait dans d'autres domaines de l'ingénierie, comme le montrent les enquêtes du Standish Group (www.standishgroup.com)⁹ (tableau 13.1).

	1994	1996	1998	2000
Succès	16 %	27 %	26 %	28 %
Échec	31 %	40 %	28 %	23 %
Dépassement	53 %	33 %	46 %	49 %

TAB. 13.1 – *Résultat des projets d'après les enquêtes du Standish Group*

On peut conserver en mémoire les ordres de grandeur suivants : un quart seulement des projets réussissent, un quart échouent complètement et sont abandonnés, une moitié aboutissent mais au prix d'un important dépassement (budget ou délai multipliés par deux ou trois).

Parmi les facteurs de succès, le premier est d'après le Standish Group l'implication des utilisateurs, qui expriment clairement leurs objectifs et formalisent leurs exigences ; le second est l'implication des dirigeants dans le soutien au projet ; parmi les causes d'échec, la première est le manque de clarté des besoins, la seconde la versatilité des spécifications en cours de réalisation.

Lorsque les dirigeants lancent un projet, ils n'ont pas ces proportions en tête et l'échec les surprend toujours. Quand un projet est en péril, ils mettent du temps à percevoir le danger et taxent de pessimisme ceux qui tentent de les alerter : les Cassandres sont mal vus. Parmi les projets qui aboutissent hors délai et hors budget, certains n'auraient été sans doute jamais été lancés si l'on avait su ce qu'ils allaient coûter.

9. L'échantillon enquêté ne comprend que des entreprises américaines, mais il est vraisemblable que les entreprises françaises ne font pas mieux.

L'examen des causes de succès et d'échec est instructif : *la plupart des échecs proviennent non de l'informatique elle-même mais de la maîtrise d'ouvrage, en englobant sous ce terme à la fois les dirigeants et les concepteurs des métiers*. La technique informatique permet beaucoup de choses et ses possibilités sont plutôt sous-utilisées. Même si la réalisation d'un projet peut comporter des errements, la compétence des SSII, des informaticiens de l'entreprise est en général suffisante pour arriver au succès.

C'est pourquoi la *professionnalisation de la maîtrise d'ouvrage* est aujourd'hui une priorité. Elle modifie les rapports avec l'informatique et déplace la frontière des responsabilités, ce qui contrarie les informaticiens dans un premier temps mais ils en voient bientôt les avantages.

Le professionnalisme de la maîtrise d'ouvrage permet d'éviter des défauts classiques et souvent simultanés : expression de besoin confuse ; priorités non définies ; demande inflationniste et versatile ; désir (caché) de régler des problèmes « politiques » à travers le système d'information ; insouciance envers la conduite de projet ; refus de prendre en considération les contraintes techniques, physiques de la plate-forme informatique (Bloch [15]).

Les entreprises parlent volontiers de leurs succès mais elles ne font pas de publicité autour de leurs échecs. Seuls sont connus ceux qui sont assez retentissants pour susciter des commentaires dans la presse. Tous s'expliquent par la conjonction de quelques-uns des défauts ci-dessus :

- les difficultés rencontrées lors du lancement du système *Socrate* par la SNCF sont dus au volontarisme de la direction qui, pour des raisons d'image et malgré les avertissements des informaticiens, a voulu le mettre en œuvre avant qu'il ne soit au point ;

- les difficultés rencontrées par l'informatique de la *bibliothèque François Mitterrand* ont été causées par la volonté de mettre les bibliothécaires au pas, ce qui a incité à construire le système sans consulter ses futurs utilisateurs et a suscité des impropriétés manifestes ;

- le *Ministère de la Justice* a confié à des magistrats la maîtrise d'ouvrage du système d'information des tribunaux : leur inexpérience a provoqué des errements coûteux ;

- le projet *Géode* de l'ANPE a été poursuivi jusqu'à la phase pilote, puis abandonné, alors que les difficultés rencontrées lors de la réalisation (dépassement des délais, lenteur du débogage, problèmes de performance) constituaient autant de signaux d'alarme.

Il arrive aussi qu'un système d'information qui fonctionne ne puisse pas rendre les services que l'on pourrait en attendre, en raison d'une définition malencontreuse résultant d'*a priori*s politiques : il en est ainsi de l'utilisation du système de facturation des *opérateurs télécoms* à des fins de marketing, l'objet de la facturation étant non le client mais la ligne.

Le Standish Group cite deux *failure stories* et deux *success stories* également instructives :

- *California DMV* (« California Department of Motor Vehicles ») a lancé en 1987 un projet pour la gestion des permis de conduire. Il a été arrêté en 1993 après avoir coûté 45 millions de dollars. Ni les dirigeants, ni les utilisateurs n'étaient motivés et les spécifications étaient confuses.

- *American Airlines* : un projet de réservation de chambres d'hôtel et de location de voitures a été abandonné en 1994 après avoir coûté 165 millions de dollars. On explique l'échec par le trop grand nombre de partenaires, l'imprécision et la versatilité des spécifications, le manque d'implication des utilisateurs.

- Les *hôtels Hyatt* ont par contre réussi leur projet de système de réservation en impliquant les utilisateurs, avec le soutien des dirigeants, et en fournissant des spécifications claires selon une définition modulaire.

- *Banco Itamarati* : cette banque brésilienne a monté un système efficace de relation avec ses clients grâce à une équipe de maîtrise d'ouvrage professionnelle qui travaillait en bonne coopération avec les informaticiens.

* *

La professionnalisation d'une maîtrise d'ouvrage réside essentiellement dans son aptitude à *urbaniser* le système d'information, puis à *modéliser* les processus ; le plan d'urbanisme étant un modèle de masse, le maître mot est *modélisation*.

Lorsque le modèle du système d'information est de bonne qualité, sobre, clair, stable, la maîtrise d'œuvre peut travailler dans de bonnes conditions. Certes la qualité du modèle n'est pas condition suffisante du succès : il faut aussi que la conduite du projet soit convenable, que la recette et le déploiement soient réussis ; mais une maîtrise d'ouvrage capable de bien modéliser l'entreprise sera, si on ne lui oppose pas d'obstacles politique, capable de réaliser convenablement les autres opérations.

Le langage UML (voir page 323) se présente aujourd'hui comme la *lingua franca* de la maîtrise d'ouvrage. Il fournit son vocabulaire à la modélisation ; il facilite la consultation des experts du métier, la concertation avec la maîtrise d'œuvre, la formalisation du besoin. Il aide à rendre naturelle, au sein du métier, la démarche d'abstraction sans laquelle il ne peut pas exister de système d'information mais qui constitue parfois le principal obstacle intellectuel. Il permet aussi, à condition que la présentation du modèle soit convenable, une validation authentique du formalisme du système d'information par les dirigeants.

Cependant le langage UML ne peut être utilisé que si l'on a au préalable mis en forme les processus de travail du métier considéré ; il ne peut être efficace que si l'expression de besoin est pertinente, sobre, cohérente ; en retour, l'élaboration du modèle permet de repérer les défauts de cohérence, d'exprimer l'exigence de sobriété : elle concourt donc à la qualité de l'expression du besoin.

Lorsque les entreprises auront compris à quel point le succès des projets informatiques dépend de la professionnalisation de la maîtrise d'ouvrage, et auront mis en place l'organisation nécessaire le taux d'échec des projets informatiques ne sera pas supérieur à ceux que l'on rencontre dans le bâtiment ou l'ingénierie industrielle.

Nous allons suivre les étapes de la réalisation d'un projet pour donner sur chacune non pas une méthode, mais des points de repères et des priorités.

13.4.1 La première expression de besoin

Certaines personnes agissent comme si les spécifications fonctionnelles devaient être écrites sous la dictée des utilisateurs. Cette démarche provoque une inflation de fonctionnalités : *a posteriori* on découvrira que les utilisateurs n'utilisent que 20 % au plus de ce qu'ils ont demandé.

Où est l'erreur ? C'est que l'on a confondu *besoin* et *demande*. Les utilisateurs expriment une demande, *mais elle ne correspond pas à leurs besoins*. Il n'y a pas là de paradoxe : ce fait d'expérience qui s'éclaire si l'on se réfère à la vie courante.

Supposons que vous vouliez faire construire une maison. Il faut qu'elle réponde à vos besoins ; ces besoins, ce sont les vôtres et non ceux d'un client standard et vous en êtes le seul porteur légitime. Mais cela ne veut pas dire que vous sachiez les exprimer, encore moins que vous sachiez s'il sera possible de les satisfaire. Votre demande est maladroite car rien ne vous a préparé à résoudre un problème d'architecture.

Michel Morel, architecte, m'a montré le carnet où il avait noté les étapes de sa conversation avec un client. Celui-ci avait acheté un terrain en Provence et voulait y faire construire une résidence secondaire. « Quel type de maison voulez-vous ? » demanda Morel. Le client dessina alors la figure 13.10, qui se passe de commentaire.

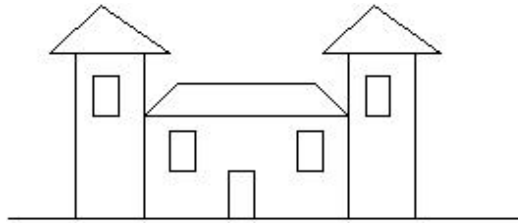
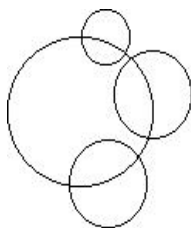
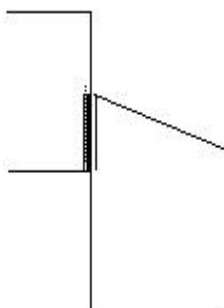


FIG. 13.10 – *Image de la demande*

Morel rangea ce dessin sur un coin de la table et dit : « Parlez-moi de vous, de votre famille, de la façon dont vous vivez ». Alors le client s'explique : il a cinquante ans, sa seconde femme a vingt-cinq ans de moins que lui, il a un fils de vingt ans qui vit avec sa première femme mais passe de temps à autre quelques jours chez lui. Il aime à s'isoler pour réfléchir, travailler ou lire ; sa femme mène une vie plus mondaine, reçoit souvent des amis etc.

Pendant que le client parle, Morel dessine des ronds représentant des espaces : salle de séjour, bureau, chambre à coucher, salle de bains, chambres d'amis etc. (figure 13.11).

Le client aimerait à s'isoler pendant les réceptions de sa femme, mais pas totalement ; Morel dessine un bureau dont le plancher est plus bas que le plafond de la salle de séjour ; il installe entre les deux une fenêtre que le client pourra ouvrir ou fermer à sa convenance, ou encore masquer par un rideau (figure 13.12).

FIG. 13.11 – *Mise en forme du besoin*FIG. 13.12 – *Détail de la spécification*

Ainsi pendant la conversation un schéma s'élabore. Morel a par la suite construit une maison qui répondait aux besoins du client, et celui-ci en est ravi - mais elle ne ressemble en rien au dessin puéril par lequel il avait d'abord exprimé sa « demande ».

Les MOAO se trouvent, devant l'utilisateur du système d'information, dans la même position que l'architecte face à son client. « Je voudrais automatiser les envois de courriers et disposer d'un système permettant de savoir, pour chaque client, ce qu'il nous a écrit et ce que nous lui avons répondu. Ce système doit me fournir des histogrammes montrant la distribution des délais de réponse, me permettre d'automatiser l'envoi de lettres de relance aux clients qui n'ont pas répondu etc. » : cela, c'est une demande. Il faut la poser sur un coin de la table et dire à l'utilisateur : « que voulez-vous faire, quel est le problème que vous cherchez à résoudre en automatisant la gestion du courrier ? » On pourra ainsi remonter de la demande au besoin et aider le client à exprimer celui-ci. Chemin faisant, on lui donnera l'occasion de tirer parti de possibilités auxquelles il n'avait pas pensé. On élaguera aussi, dans la formulation de la demande, les fonctionnalités dont on peut présumer qu'elles ne serviront à rien.

Il est vrai que des clients au caractère impérieux refusent de se prêter à ce jeu. « Je sais ce dont j'ai besoin, disent-ils, et je n'ai pas envie de vous raconter ma vie. Faites donc ce que je vous demande ». Il faut alors leur dire, avec respect mais fermement, que ce n'est pas ainsi qu'ils seront le mieux servis.

Pourquoi nos systèmes d'information ne sont-ils pas naturellement sobres? C'est pour une raison philosophique d'autant plus profonde qu'elle est implicite, et aussi pour une raison institutionnelle.

Toute représentation simple répugne à ceux qui la considèrent comme une violence faite à la complexité du réel. Ils lui préféreront une représentation compliquée (voir page 219); bien sûr, s'agissant d'une représentation, celle-ci sera simple comparée au réel qu'aucune représentation n'épuise, mais sa complication évoquera ou plus exactement *singera* la complexité du réel.

Cette esthétique de la complication se manifeste par des réactions vives, voire douloureuses, devant toute tentative de simplification ou de choix dans les priorités; la phrase clé est « Ce n'est pas si simple ». Toute tentative de classification, hiérarchisation, distinction entre des couches logiques que l'action entremêle mais que le discernement doit séparer, suscite cette objection.

La loi des 80-20 (80 % de l'utilité résident dans 20 % des fonctionnalités) invite pourtant à la sélectivité. Il vaudrait mieux, bien sûr, laisser tomber les 80 % qui ne servent à rien; mais l'amoureux de la complication s'y refuse. « Les utilisateurs le demandent », voici sa deuxième phrase clé.

Oui, les utilisateurs demandent beaucoup de choses; mais attendent-ils qu'on les suive à la lettre? non, car l'utilisateur préférera que l'on fasse dans sa demande un tri intelligent et que l'on prépare le produit simple qui satisfera l'essentiel de ses besoins.

Les concepteurs confondent trop souvent, insistons-y, « demande » et « besoin ». Le besoin, réalité profonde enfouie dans la conscience et la pratique de l'utilisateur, ne s'exprime pas directement; il se traduit par une demande explicite mais elle est infidèle comme toute traduction: la formulation de la demande dépend de l'idée que l'utilisateur se fait du possible technique et mêle sans tri ni classement l'indispensable, le nécessaire, l'utile, le commode, le superflu et le futile.

Supposons que pour confectionner l'expression de besoins on ait rassemblé un groupe d'utilisateurs. Son animation devra passer par les phases suivantes:

Phase 1: recueillir les demandes de la façon la plus complète possible, car il faut lancer un grand filet aux mailles serrées pour être sûr de ne rien négliger d'important; remplir ainsi des affiches en papier qui couvriront les murs de la salle de réunion; insister pour être certain de ne rien avoir oublié.

Phase 2: demander aux utilisateurs d'identifier, parmi les fonctionnalités qui ont été notées, celles qui sont *indispensables* et que l'on soulignera en rouge sur les affiches; insister pour qu'elles soient *vraiment* indispensables en posant des questions comme « êtes-vous sûrs que l'entreprise s'arrêtera si l'on ne fait pas ainsi? » ou « êtes-vous sûrs qu'on ne peut pas travailler sans cela? »: les fonctionnalités indispensables représenteront de 5 à 15 % de celles qui ont été recensées dans la phase 1.

Phase 3: dire enfin aux utilisateurs « La V1 fournira toutes les fonctionnalités indispensables, les autres seront fournies par la V2 ».

Le secret du succès - mais il ne faut pas le dire, puisque c'est un secret - c'est qu'il ne faudra jamais faire cette V2 là. L'histoire de l'informatique

abonde en bons produits qui furent détruits par une V2 où l'on a voulu introduire tout ce qui avait été négligé par la V1. Les versions ultérieures - il y en aura, bien sûr - ne devront pas tenter de satisfaire « toute la demande » qui avait été recensée initialement, mais seulement les besoins nouveaux survenus entre temps et toujours sous la même contrainte de sélectivité.

Un système d'information est fragile et instable, sujet à des pannes, des incidents, il est difficile de le faire évoluer pour l'adapter à un environnement changeant. Sa fragilité est à peu près proportionnelle au carré de la taille des programmes ou du nombre des fonctionnalités : elle croît donc vite avec la complexité. Plus un système est sobre, plus il sera solide, moins il connaîtra d'incidents, mieux on pourra le faire évoluer. La sobriété facilite non seulement la conception, mais l'exploitation et l'évolution, et l'utilisateur y gagne.

* *

La répugnance envers l'abstraction, explique les symptômes de souffrance que constate celui qui invite à la sobriété. Les réactions de colère, elles, s'expliquent par des raisons institutionnelles.

Conseiller une solution simple, suggérer une priorité, c'est énoncer un avis clair et donc prendre le risque d'un démenti. Il y faut du courage : si rechercher la simplicité c'est rendre service à l'entreprise, ce n'est pas se rendre service à soi-même.

Les hommes de pouvoir et de peur (c'est la même chose) ne répondront jamais à celui qui propose une simplification en se plaçant sur le même terrain pratique que lui. Ils ne diront pas « examinons si cette simplification-là est la plus pertinente », mais « ce n'est pas si simple », « il faut tout de même être sérieux », « nous devons répondre à la demande des utilisateurs », « je ne suis pas convaincu », « j'ai peur que... » etc. Ces phrases procurant une prime de crédibilité, ceux qui les énoncent se rengorgent en exhibant le « goitre du dirigeant » qui donne à la voix un son grave.

Être simple de façon judicieuse suppose une clarté de vue, un coup d'œil et un sang froid que ne peuvent pas avoir ceux qui se sentent dépassés par leurs responsabilités. Ils auront le réflexe malheureux de l'apprenti motocycliste qui se raidit par crainte de tomber en s'inclinant dans les virages, et tombe fatalement dans le fossé extérieur. L'homme que sa mission effraie crée d'instinct les conditions de l'échec. Croyant le succès impossible il préfère, « foutu pour foutu », se mettre du côté des fortes probabilités - et l'un des procédés les plus sûrs pour susciter l'échec est de refuser, sous prétexte de sérieux, tout ce qui pourrait simplifier la tâche. Le simplificateur provoque des réactions d'une violence qui surprendrait si l'on ne savait l'interpréter : les personnes en proie à la panique considèrent celui qui leur indique une issue comme leur pire ennemi.

Pour faire progresser la sobriété, il faut en lancer la mode. Les ingénieurs, dans leur sérieux professionnel, obéissent à une esthétique qui guide en profondeur leurs choix et leurs attitudes. Si la mode est à la complication, si la phrase « ce n'est pas si simple » suscite le respect, ils sont désarmés. Mais si cette phrase déconsidère celui qui la prononce, si la mode est à la sobriété,

si celui qui propose une simplification est écouté avec intérêt, alors on verra les projets se clarifier, les priorités s'exprimer, le langage s'épurer du jargon par lequel les corporations cherchent à se protéger, enfin la communication s'établir.

* *

L'expression de besoin est le premier document que produise une maîtrise d'ouvrage pour amorcer un projet nouveau. En cours de projet, les soucis de la réalisation risquent de faire perdre de vue les intentions initiales : la relecture de l'expression de besoin permettra alors un utile retour aux sources. C'est pourquoi elle doit rester accessible pendant toute la durée du projet. Comme il s'agit d'un texte court, on aura intérêt à le reproduire intégralement en tête des autres documents relatifs au projet (étude OFR, FSM, spécifications générales etc.).

On peut décliner les qualités que doit posséder une expression de besoins à partir de celles que l'on attend du système d'information (voir page 255) : elle doit être (a) pertinente, (b) sobre, (c) complète, (d) claire à la lecture.

La *pertinence* se juge relativement à l'action qu'il s'agit de conduire. L'expression de besoin doit contenir une description de cette action : il faut dire à *quoi* servira ce que l'on demande, à *qui* cela sera utile, situer le besoin dans le processus de travail ainsi que par rapport à la stratégie de l'entreprise et à ses priorités, évoquer enfin ce qui se passerait si le besoin n'était pas satisfait.

La *sobriété* concerne la façon dont on envisage de satisfaire le besoin. Un système d'information sobre est celui qui satisfait raisonnablement les besoins après un arbitrage entre coût et satisfaction de l'utilisateur : il laisse donc délibérément non satisfaits des besoins non prioritaires. Quelle serait la réponse minimale au besoin ? Une version réduite serait-elle possible ? Quelles sont les fonctionnalités auxquelles il a été jugé préférable de renoncer ?

La *complétude* n'est pas la fuite dans le détail (ce serait contraire à la sobriété), mais l'assurance que l'on n'a rien oublié d'important tant en ce qui concerne le projet lui-même que son insertion dans le système d'information. Il faut décrire les relations entre le projet et le système d'information, situer les points de coordination nécessaires, présenter enfin la méthode utilisée pour établir l'expression de besoins (consultations, validations).

La *clarté* est indispensable pour un document qui doit circuler entre plusieurs mains, nourrir des arbitrages, et qui restera pendant toute la vie du projet la source à laquelle on se réfère pour revenir à l'intention initiale.

L'expression de besoin sera matériellement un texte en langage naturel de deux à trois pages au plus, complété par quelques données fournies en page de couverture : identité du chef de projet à la maîtrise d'ouvrage ; identité du maître d'ouvrage stratégique (directeur du métier), du maître d'ouvrage délégué, de l'auteur de l'expression de besoin ; indication sur la priorité ; calendrier de réalisation souhaité, phasage éventuel ; estimation qualitative de l'importance du projet ; indications qui permettront à la maîtrise d'œuvre d'inférer la (ou les) solution(s) possible(s) et d'estimer leur

coût (volumétrie, contraintes de performance, nombre d'écrans, processus ou composants ou applications concernés).

13.4.2 Étude Opportunité, Faisabilité, Risques

On appelle « Étude OFR » (ou encore « Étude préalable ») l'étude fournie aux dirigeants pour préparer la décision de lancement ou de non lancement d'un projet. Elle est réalisée sous la responsabilité de la maîtrise d'ouvrage opérationnelle, avec la contribution du MOAD et de l'informatique. Elle est validée par le MOAS qui la présente au CSSI si l'enjeu le justifie.

La décision de lancer l'étude OFR est prise après sélection parmi les expressions de besoin ; l'étude OFR ayant un coût non négligeable son lancement n'a rien d'automatique : certaines expressions de besoin recevront donc une réponse négative avant même que l'on ne la réalise.

Le plan de l'étude OFR comporte quatre parties :

Opportunité : évaluation des besoins que le projet vise à satisfaire (adaptation à une contrainte externe, baisse du coût de production, lancement d'un nouveau produit, gain de part de marché, accroissement du chiffre d'affaires, le tout aboutissant à une hausse du résultat) et du gain pour l'entreprise. L'évaluation du gain doit résulter d'un raisonnement différentiel c'est-à-dire de la comparaison de deux scénarios « avec le projet » ou « sans le projet » (il ne faudrait pas attribuer au projet un gain tendanciel qui se serait produit même si on ne le réalisait pas).

L'étude d'opportunité est fondée sur les éléments déjà fournis par l'expression de besoin (priorité, importance stratégique, urgence etc.) qu'elle approfondit et précise en les quantifiant autant que possible. Cette partie de l'étude est réalisée par la maîtrise d'ouvrage.

Faisabilité : alors que l'étude d'opportunité ne fait que préciser l'expression de besoin, l'étude de faisabilité est entièrement nouvelle : il s'agit de dire si, et comment, le produit est réalisable en proposant un ou plusieurs scénarios de solution, en évaluant la disponibilité des ressources (humaines, techniques, matérielles) nécessaires pour l'investissement initial et pour le fonctionnement et en estimant leur coût (Printz [166] et [167]), là aussi de façon différentielle. Cette partie de l'étude est faite par la DSI.

La faisabilité doit aussi être évaluée du côté de la maîtrise d'ouvrage : disponibilité des compétences, effets sur les processus et l'organisation, cohabitation entre le nouveau produit et les anciens produits.

Économie : en confrontant le résultat et le coût différentiels anticipés, et en tenant compte de la durée de vie anticipée du produit, on évalue l'apport du projet en termes de rentabilité (TRI, taux de rentabilité interne) et de valeur actuelle nette (VAN).

Les TRI et VAN que l'on peut estimer à ce stade sont approximatifs. Il faut les prendre en considération, puisque ce sont les seules estimations dont l'entreprise dispose, mais il faut être conscient de leur imprécision.

Risques : les risques concernant le projet sont de divers ordres :

- risques associés à la réalisation : risques concernant les coûts et les délais ; risques relatifs à la disponibilité des éléments de la solution prévue ;
- risques associés à la mise en œuvre : incertitudes relatives à l'acceptation par le marché et par l'organisation de l'entreprise.

La probabilité associée à chacun des risques est évaluée, ses effets sont décrits, les mesures à prendre s'il se manifeste sont indiquées.

* *

La réalisation d'une étude OFR consomme du temps des experts internes à l'entreprise, éventuellement du temps de consultant externe. Il serait un peu ridicule de consacrer à cette étude un effort sans proportion avec l'importance du projet ou de préparer longuement et lourdement une décision *a priori* évidente.

L'étude OFR sera donc plus légère, voire réduite à quelques paragraphes, pour les projets de petite taille et de faible importance ou pour ceux dont le lancement est de toute façon obligatoire.

Le dosage de l'effort à consacrer à l'OFR est affaire de bon sens et on ne peut pas le définir par un pourcentage du coût anticipé du projet : il peut arriver qu'un projet de petite taille, mais fonctionnellement important, suscite une controverse. Il peut alors se justifier de consacrer à l'étude OFR un montant du même ordre de grandeur que le coût du projet car l'étude fera progresser la maîtrise du système d'information par l'entreprise.

Le succès de l'étude OFR réside dans la justesse de la décision qu'elle prépare, que celle-ci soit de lancer ou de refuser le projet. Cette justesse ne peut être obtenue avec sûreté que si les parties concernées ont pu s'exprimer à l'occasion de l'étude : s'il existe des opposants, leur position doit avoir été présentée de façon claire et équitable. Il faut aussi que le positionnement du projet par rapport au système d'information et à d'autres projets ait été perçu et que l'on ait examiné des alternatives : « Que se passerait-il si l'on allait plus loin dans le sens du projet ? que se passerait-il si on ne le réalisait pas, ou si on réduisait ses ambitions ? » Il faut enfin que les dirigeants concernés aient pu se faire une idée du contenu du projet et exprimer leurs avis.

L'étude OFR est organiquement reliée aux autres étapes de la démarche : elle reprend et enrichit l'expression de besoin qui était en amont ; en aval, elle alimente le suivi des risques qui se poursuivra pendant la durée du projet ; les éléments économiques qu'elle contient serviront de référence pour évaluer la réussite du projet. La préparation de l'OFR nécessite une coopération équilibrée entre la MOA et la MOE et une validation par les responsables des deux bords.

La suite de l'étude OFR, c'est soit le renoncement au projet, soit le lancement de la démarche de modélisation ; mais avant cela il faut avoir précisé les rôles des acteurs du projet.

13.4.3 Fiche de synthèse de mission

La « fiche de synthèse de mission » (FSM, appelée aussi « contrat de projet ») indique le partage des responsabilités et des pouvoirs entre les parties impliquées dans la réalisation du projet. Elle précise les engagements pris par chacune en termes de résultat, de démarche et de moyens ; elle décrit l'organisation de la réalisation, ses délais, ses balises. Elle sera pour les parties le document contractuel de référence pendant la durée du projet. L'élaboration de la FSM est aussi l'occasion d'un dialogue qui leur permettra de comprendre leurs enjeux respectifs et d'élaborer un langage commun.

La FSM explicite un accord entre la maîtrise d'ouvrage et la maîtrise d'œuvre sur le périmètre du projet, les priorités, le découpage en lots, le calendrier de réalisation, les points critiques et les risques sur lesquels l'effort doit se concentrer. Elle décrit la façon dont la MOA et la MOE devront communiquer : la MOA devra fournir les expertises nécessaires à l'élaboration des spécifications ; la MOE devra fournir un reporting fidèle sur l'avancement du projet, les difficultés rencontrées, et négocier avec la MOA les éventuels arbitrages (sacrifier des fonctionnalités pour tenir les délais etc.).

La FSM précise la composition et la périodicité des divers comités (comité d'avancement, comité de pilotage, comité de direction), la nature des documents préparés pour ces comités, la structure de la documentation mise à la disposition des participants au projet.

Il importe que ces décisions soient prises « à froid », avant que les premières difficultés ne se manifestent - car alors tout serait bloqué.

La FSM désigne enfin deux acteurs clés du projet : le directeur du projet (ou chef de projet MOA) ; le chef de projet maîtrise d'œuvre. Le directeur de projet est le garant de la conformité du montage du projet par rapport aux attentes de la MOA ; le CP MOE est responsable de la réalisation des livrables impliqués par le projet.

Ces deux personnes sont les représentants privilégiés respectivement de la MOA et de la MOE tant lors de l'élaboration de la FSM que lors de la réalisation du projet. Ils organisent, chacun pour sa part, l'intervention des acteurs impliqués : côté MOA, experts métiers, représentants des utilisateurs et MOAS ; côté MOE, responsables des études, de la production, du réseau télécoms et DSI. Ces acteurs valident et signent la FSM qui engage ainsi non seulement la MOA et la MOE considérées dans leur ensemble, mais des niveaux de responsabilité plus spécialisés.

La suite du projet réside dans la modélisation (se reporter à la page 445) ; à l'issue de la modélisation s'engage la réalisation du projet proprement dite.

13.4.4 Conduite de projet

Le but de la conduite de projet est d'assurer la relation entre l'entreprise et le ou les fournisseur(s) pendant la réalisation en ayant pour but unique la réussite du projet - c'est-à-dire la mise en exploitation d'un produit conforme aux spécifications et dans les conditions prévues de délai et

de coût. Nous avons vu page 453 que la plupart des projets informatiques échouaient en cours de route, ou connaissaient des dépassement de coût et de délai considérables : ce but à la formulation si simple n'est donc pas facile à atteindre.

Le plus souvent, l'application sera réalisée par une SSII, mise en exploitation par la DSI et utilisée par un métier de l'entreprise. C'est ce dernier (maîtrise d'ouvrage) qui est donc le client final de la SSII ; il lui revient donc, en bonne logique, d'assumer la responsabilité de la direction du projet ainsi que la vérification du service fait et la gestion du budget.

Dans les faits, cependant, ces diverses tâches sont souvent déléguées à la DSI par application du slogan favori des directeurs généraux : « tout ça, c'est de l'informatique ». Il en résulte parfois des dommages, les impératifs techniques de la plate-forme informatique recevant plus de poids dans les arbitrages que les besoins du métier. Il faut dire aussi que les maîtrises d'ouvrage se sont rarement dotées des compétences nécessaires et qu'il leur est trop commode, trop tentant de faire porter par la DSI la responsabilité d'échecs si fréquents.

* *

L'organisation qui convient dépend de la taille du projet, de la place du processus concerné parmi les métiers de l'entreprise (est-il propre à un métier, traverse-t-il plusieurs métiers?), de l'importance du projet en regard de la stratégie de l'entreprise etc. Elle doit donc être définie au coup par coup et il n'existe pas de formule générale s'imposant à tout projet.

Nous allons ci-dessous décrire des éléments que l'on rencontre le plus souvent et signaler les pièges qui peuvent se présenter.

Directeur de projet

Le but du projet est de fournir un produit utile à une maîtrise d'ouvrage. Il est donc naturel que la direction du projet soit confiée à un représentant de la maîtrise d'ouvrage, dûment mandaté par le maître d'ouvrage stratégique (qui sera, selon la nature du projet, le directeur d'un des métiers de l'entreprise ou le directeur général lui-même). Le directeur de projet « coiffe », dans la structure de projet, le maître d'ouvrage opérationnel et le chef de projet maître d'œuvre. Il est assisté par un pilote de projet chargé du reporting et du suivi de l'exécution des décisions.

Il revient au directeur de projet de définir dès le début du projet avec ces trois responsables le contour de leurs missions, l'échéancier des travaux que chacun doit réaliser, la nature de leur coopération.

Il faut que son mandat constitue une vraie délégation de responsabilité : un directeur de projet dont les décisions sont fréquemment désavouées par le mandataire devient rapidement un « zombie », personnage falot dont le rôle n'est pas même symbolique.

Il faut aussi que ce directeur puisse *réellement* diriger le projet, qu'il dispose pour cela d'informations sur la qualité de la réalisation, la tenue des délais, l'avancement des travaux, la situation du budget (voir ci-dessous le paragraphe consacré au reporting). Il arrive souvent que la partie financière du projet soit gérée par la DSI qui soit par désordre, soit par souci

d'indépendance, ne fournit pas d'information au directeur de projet. Il est donc préférable - même si c'est une disposition rarement prise - que le directeur de projet soit investi de la responsabilité du suivi du contrat et de la certification du service fait et que ce soit lui qui déclenche le paiement des factures de la SSII.

Souvent le directeur de projet, estimant son affaire déjà bien assez difficile, cherche à se libérer des contraintes de cohérence du système d'information et à s'isoler du reste de l'entreprise. Rien n'est plus difficile que d'organiser une réunion entre plusieurs directeurs de projet ! Or un manque de cohérence du système d'information serait dommageable pour l'entreprise. Il revient à la coordination des maîtrises d'ouvrage (la MOAD auprès du directeur général) d'établir des passerelles entre les divers projets et d'animer le dialogue entre les directeurs de projet malgré leurs réticences.

* *

Tout suivi de projet est scandé par les réunions de divers comités (voir ci-dessous). L'expérience, fort pénible, montre que ces réunions sont souvent longues et confuses, sans ordre du jour, durée limite ni compte rendu. Or la conduite du projet dépend pour une bonne part de la qualité de ces réunions. Il faut respecter quelques règles :

- il est interdit en réunion de critiquer une personne absente ou une entité qui n'est pas représentée car il faut avant tout respecter le droit de réponse ;
- l'ordre du jour de la réunion doit être établi avant la réunion, éventuellement complété en début de réunion, et rigoureusement suivi en proscrivant les digression et coqs-à-l'âne ;
- les tâches décidées en réunion doivent être affectées à une personne ou une entité (ne jamais dire « on va faire cela » à la cantonade, sans désigner qui devra le faire) ;
- les décisions doivent être notées par le pilote du projet et leur exécution doit faire l'objet d'un pointage lors des réunions suivantes.

Outre son rôle purement technique et professionnel, le directeur de projet doit se soucier de « l'ambiance » du projet. Tout grand projet comporte des moments de crise, de doute, voire de désespoir et dans ces moments-là les responsables cherchent moins à faire leur travail qu'à faire porter par d'autres la responsabilité d'un échec qui leur semble inévitable.

Il faut alors du talent pour apaiser les conflits, restaurer ou maintenir l'espoir de succès : le directeur de projet doit savoir maîtriser l'expression de ses sentiments. En général il devra afficher de la sérénité et se comporter avec bonhomie envers les personnes ; en de rares occasions, il devra aussi manifester de l'inquiétude et se mettre en colère. La gestion de la psychologie collective d'un projet est un art aussi délicat que la diplomatie.

Comité directeur

Le comité directeur d'un grand projet, mensuel ou trimestriel selon les cas, réunit le directeur de projet, le DSI, le directeur général, les directeurs concernés par le projet et les experts invités (y compris ceux qui représentent le fournisseur, au moins pour une partie de la réunion). Son but est de

vérifier que le déroulement du projet est conforme aux objectifs stratégiques de l'entreprise notamment en ce qui concerne les délais et les coûts.

L'expérience montre que ces réunions sont souvent édulcorées, personne n'osant en présence du directeur général et devant des directeurs plus ou moins anxieux émettre la moindre alarme ni poser une question de simple bon sens.

Il revient au directeur général de faire en sorte que le comité directeur ne soit pas un de ces lieux où personne n'ose dire la vérité : s'il fait taire celui qui émet un signal d'alarme, les réunions seront vides de contenu.

Comité de pilotage

Le comité de pilotage est hebdomadaire ou mensuel selon l'importance du projet. Il réunit le directeur du projet, les chefs de projet MOA, MOE et le pilote du projet, ainsi que les experts invités. Périodiquement, le fournisseur lui-même est invité. C'est la réunion principale pour la conduite de projet, c'est là que se prennent les décisions essentielles au vu des reportings et des alarmes émises par les parties. La qualité de ces réunions dépend du prix que le directeur de projet attache à la rigueur du suivi de la réalisation et au respect des décisions.

Comité d'avancement

Le comité d'avancement est hebdomadaire ; il réunit les chefs de projet MOA, MOE et le pilote du projet, ainsi que les représentants du fournisseur. Il a pour but de repérer et de régler les problèmes d'ordre matériel ou organisationnel que pose la réalisation : disponibilité des locaux, serveurs et postes de travail ; *turn over* des personnes ; tenue des délais de production des livrables ; traitement des alarmes émises lors des comités précédents.

* *

Le projet doit faire l'objet d'un compte rendu régulier (« reporting ») qui permette d'évaluer sa situation. Les reportings présentent souvent des défauts :

- trop lourds, ils sont difficiles à lire et l'information utile est noyée dans un fatras de nombres et de graphiques ;
- remplis à la va-vite, ils comportent des incohérences qui sautent aux yeux : on passe plus de temps à se les faire expliquer qu'à traiter les problèmes ;
- ils présentent les derniers événements sans les rattacher à l'historique du projet, ce qui rend difficile leur interprétation : si par exemple le fournisseur dit « la livraison du lot 3 sera reculée de trois semaines », le sens de cette phrase n'est pas le même selon que cette échéance a déjà été reculée plusieurs fois ou non ;
- ils obéissent de façon pédante au formalisme d'une « méthode qualité » sans faire le tri entre l'important et l'accessoire (c'est là un symptôme d'incompétence, ou pis de mauvais esprit).

Les personnes engagées dans le projet jugent naturellement le travail de production plus important que le reporting, qu'elles considèrent comme une corvée. C'est ce qui les pousse à le remplir à la va-vite sans se soucier de sa lisibilité ni de son exactitude. On voit ainsi apparaître des erreurs manifestes

(comme un taux de réalisation de 150 % ou un échéancier prévisionnel qui ne tient aucun compte des retards enregistrés dans le passé).

Pour que le reporting soit sérieusement rempli, il faut (1) qu'il porte la signature d'une personne chargée de vérifier l'exactitude des informations qu'il contient, et qui sera responsable des éventuelles erreurs ou lacunes ; (2) que les exécutants sachent qu'à l'occasion certains reportings peuvent remonter jusqu'au sommet de l'entreprise et y susciter des décisions.

Il est salubre de faire parvenir au directeur général une sélection des reportings. Un directeur général conscient de l'importance du système d'information doit se faire communiquer les reportings mensuels des principaux projets et on doit lui montrer ceux des projets de moindre importance, mais qui rencontrent des difficultés et sur lesquels on risque de devoir lui demander des décisions.

Pour un suivi de projet efficace

Le maître d'ouvrage doit disposer d'une documentation lui permettant de percevoir clairement le déroulement du projet ; elle doit être d'une lecture facile.

Pour que le client puisse suivre efficacement un projet, il faut qu'il dispose avant que celui-ci ne soit lancé d'une description du projet comportant :

1. une vue d'ensemble du projet (2 pages) ;
2. une fiche décrivant poste par poste l'ensemble des opérations associées au projet (y compris celles non spécialement informatiques comme la formation des utilisateurs, la mise à niveau des réseaux télécoms etc.) ainsi que leurs échéances ;
3. la liste des échéances cruciales (lots) et des livrables correspondants ;
4. pour chaque lot une fiche (1 ou 2 pages) décrivant son contenu et son coût et comportant si besoin est des liens vers des documents détaillés.

La fourniture des documents détaillés est facultative. Leur utilité est surtout de permettre au fournisseur de décrire les détails techniques qu'il juge importants, mais qui n'intéressent pas le client en priorité. Le client se contentera le plus souvent des fiches succinctes décrivant les lots et livrables.

L'ensemble de cette documentation sera commodément placée sur un serveur Intranet. Lors du déroulement du projet il faut garder en mémoire sa description initiale et de visualiser les décalages survenant tant sur les échéances et les dépenses que sur le contenu des livrables. Il faut pour cela établir des graphiques et documents mis à jour périodiquement (selon les délais et l'importance du projet, il faudra prévoir une mise à jour par semaine, par mois ou par trimestre). Les deux graphiques les plus importants décrivent (1) les délais de réalisation, (2) l'évolution de la dépense.

Pour suivre les délais de réalisation on porte sur l'axe vertical les dates prévues de livraison des divers lots, et sur l'axe horizontal le déroulement du temps. Lors de chaque rapport périodique on met à jour les dates de livraison prévues : les dérapages éventuels sont alors visuellement manifestes, la courbe qui représente l'évolution des dates prévues se courbant vers le haut (il arrive que l'on voie se courber vers le haut la courbe qui représente le prochain lot, mais que le fournisseur maintienne les prévisions concernant

les lots suivants : il faut alors lui demander comment il s'y prendra pour accélérer la cadence).

Pour le suivi de la dépense, on mettra en regard l'échéancier prévu et le cumul des dépenses réalisées¹⁰, prolongé par l'estimation du montant et des échéances des dépenses futures.

Il ne suffit pas de contrôler les délais et la dépense : il faut aussi s'assurer du contenu des livrables (sinon, il serait trop tentant de tenir les délais en faisant glisser un contenu vers le livrable suivant). La mise à jour de la documentation doit indiquer en rouge (car cela doit sauter aux yeux) la date et le contenu de chacune des modifications éventuelles d'un livrable par rapport à sa définition initiale.

La fourniture initiale de la documentation décrite ci-dessus, puis sa tenue à jour, permettent au client de suivre la réalisation sans se noyer dans des détails qui ne le concernent pas. La tenue à jour de cette documentation doit faire l'objet d'une clause du contrat et sa qualité doit engager la responsabilité du fournisseur.

13.4.5 Recette

La « recette », c'est l'opération par laquelle le client reconnaît que le produit qui lui a été livré est conforme à la commande passée, qu'il est exploitable dans le système d'information de l'entreprise, enfin qu'il est opportun de le mettre à la disposition des utilisateurs. Elle est réalisée selon des procédures qui ne s'improvisent pas. Elle comporte plusieurs étapes. Enfin, elle présente des risques.

La théorie, confirmée par la pratique, enseigne qu'il est impossible de prouver qu'un logiciel ne comporte aucune erreur (cf. théorème de Gödel, page 85). La procédure de recette ne peut donc pas être exhaustive : quel que soit le soin que l'on apporte à la définition des tests on n'aura jamais la certitude que le logiciel ne comporte aucune bogue.

On peut toutefois, et cela relève du bon sens, vérifier qu'il remplit correctement les fonctions essentielles que l'on attend de lui que ce soit en termes de performances, de qualité des données ou d'interface homme-machine. La liste des tests doit être établie à froid, avant la livraison du produit ; elle porte le nom de « cahier de recette ».

La combinatoire des tests possibles étant infinie, le cahier de recette résulte d'une sélection. Certains tests sont coûteux en raison de la volumétrie ou des difficultés techniques qu'ils impliquent : le cahier de recette doit comporter la liste de tests la plus efficace pour un coût donné.

Il faut que la recette soit effectuée en partant de « vraies données » et non de données de qualité parfaite : on sait en effet qu'en informatique de gestion les données sont souvent, pour des raisons parfaitement compréhensibles,

10. Par « dépenses réalisées », on entend non les engagements, les décaissements ni les factures reçues, mais l'estimation du coût des travaux effectivement réalisés à la date d'évaluation de la dépense. On peut faire figurer dans des annexes, si on le juge utile, les engagements, le montant des factures reçues et les décaissements etc.).

de mauvaise qualité (codages imparfaits, données manquantes, vérifications négligées). C'est par rapport à cet état de fait qu'il faut évaluer la qualité du produit, et non par rapport au monde idéal où les données seraient sans défauts.

Il faut préciser le protocole selon lequel la recette sera organisée : quelles seront les tâches qui incomberont au client, celles qui incomberont au fournisseur ; quels seront les documents qu'ils devront se communiquer ; dans quel ordre les tests seront réalisés ; quels seront les seuils d'acceptation du produit. Si le protocole n'est pas défini à l'avance, le risque de conflits ou à tout le moins de malentendus entre client et fournisseur sera élevé.

On distingue deux étapes dans la recette : la « recette usine », faite par le fournisseur avant la livraison du produit, permet à celui-ci de vérifier que le produit est conforme à la commande reçue ; la « recette utilisateur » faite par le client après livraison.

Il faut que le compte rendu de la recette usine soit livré par le fournisseur en même temps que le produit : il prouvera au client que le produit a été sérieusement testé avant sa livraison et lui permettra de gagner du temps en ne refaisant pas les tests déjà réalisés par le fournisseur. Il faut prévoir la livraison de ce compte rendu dans le protocole de recette, sans quoi le client aura du mal à l'obtenir.

La recette utilisateur comporte deux étapes :

- une *recette technique*, réalisée par la DSI du client, vérifie que le produit est exploitable sur la plate-forme informatique de l'entreprise (compatibilité avec ses matériels, systèmes d'exploitation et logiciels) et que la performance physique est acceptable (volumétrie des bases de donnée et des flux de messages, délais d'affichage sur les écrans des utilisateurs, robustesse en exploitation) ;

- une *recette fonctionnelle*, réalisée par la maîtrise d'ouvrage, vérifie que le produit fournit les fonctionnalités demandées par le cahier des charges et qu'il est acceptable par les utilisateurs.

Les tests détectent des anomalies ; chaque anomalie fait l'objet d'une « fiche d'anomalie » envoyée au fournisseur qui corrige les anomalies jusqu'à convergence des tests. Il arrive parfois que la correction d'une anomalie provoque d'autres anomalies, ce qui peut obliger à refaire des tests qui avaient auparavant donné un résultat acceptable.

Il est inévitable qu'un logiciel d'une certaine importance contienne des erreurs : la présence d'anomalies au début de la recette n'a rien de scandaleux même si elle suscite toujours une tension entre client et fournisseur. Le véritable critère de qualité réside dans le délai de correction des anomalies : si ce délai est long, si les corrections suscitent d'autres anomalies de telle sorte que le nombre d'anomalies à traiter ne diminue pas, le client doit s'interroger sérieusement sur la qualité du logiciel et sur son évolutivité.

Lorsque les tests de recette ont convergé, le client prononce une « vérification d'aptitude » (VA). Il est d'usage que le fournisseur facture à ce moment là de 80 à 90 % du prix du produit (mais non la totalité, car il reste du travail à faire). Puis le produit est mis en exploitation sur un site pilote.

On détectera alors d'autres anomalies puisque le cahier de recettes ne pouvait pas être exhaustif. Elles devront elles aussi être corrigées. La convergence de ces corrections peut demander quelques mois.

Une fois corrigées les anomalies le client prononce la « vérification de service régulier » (VSR), ce qui permet au fournisseur de facturer le solde du prix du produit. Le produit peut alors être déployé sur tous les sites de l'entreprise et l'on passe à l'étape du déploiement.

* *

Toute recette présente des risques. La liste des tests que comporte le cahier de recette est inévitablement limitée : certaines anomalies n'apparaîtront donc que lors du site pilote. Il est donc préférable que le délai entre la vérification d'aptitude et la mise en exploitation soit court. Si le produit est de grande taille, et livré sous forme de lots successifs, on s'efforcera de définir ces lots de sorte que chacun soit un « module exploitable », qu'il soit possible de le mettre en exploitation sur un site pilote dès sa livraison afin d'éviter l'« effet tunnel » qui se produit lorsque la mise en exploitation ne peut se faire qu'après la réception de l'ensemble des divers lots.

Il convient, lors de l'élaboration du cahier de recette, de ne pas trop hiérarchiser les tests : on risquerait de bloquer longtemps certains tests en l'attente de la correction des anomalies détectées en amont.

Le délai nécessaire à la convergence des tests est aléatoire : on ne peut pas évaluer *a priori* la difficulté des corrections. Il faudra gérer la crise entre client et fournisseur quand ce délai semble s'allonger indéfiniment ; elle peut aboutir soit (finalement) à une livraison de qualité acceptable, soit au refus du produit.

Enfin la recette est toujours pour le client un moment délicat, puisque le produit qu'il découvre résulte à la fois des spécifications qu'il a fournies et de la réalisation qu'a bâtie le fournisseur sur la base de ces spécifications. Certaines des fiches d'anomalie seront interprétées par le fournisseur comme des demandes d'évolution et il demandera pour les corriger un avenant au contrat. Il en résultera des négociations lors desquelles la relation entre client et fournisseur risque de se détériorer.

13.4.6 Déploiement

Il faut que la mise en place sur le terrain, que l'on appelle « déploiement », ait été préparée conjointement par la MOA et la MOE avant la VSR (« vérification de service régulier », étape ultime de la recette). L'organisation du déploiement doit préciser le calendrier des actions, identifier les acteurs et préparer leur coopération y compris avec les éventuels sous-traitants.

Le but du déploiement est de faire en sorte que l'application puisse fonctionner sur les postes de travail des utilisateurs, sur les réseaux télécoms, les imprimantes, les automates qui lancent les traitements en batch et, de façon générale, sur tous les équipements que l'application utilise.

La préparation technique du déploiement dépend de l'environnement de l'application. Si celle-ci est conçue selon l'architecture client-serveur, il faut prévoir la mise en production sur le serveur et aussi un environnement de formation pour les utilisateurs.

Le chargement de la base de données (ou la migration des données de l'application ancienne) dépend de la façon dont se fera le déploiement : les méthodes ne sont pas les mêmes selon que le déploiement se fait par « big bang » ou de façon progressive.

Il faut s'assurer que les conditions nécessaires à l'exploitation sont respectées : qualification des traitements batch, dimensionnement des outils d'exploitation, formation des personnes chargées de l'exploitation et de la supervision. L'installation sur les postes de travail peut nécessiter une industrialisation si le nombre de postes à équiper est élevé. Les outils de télémaintenance et de télédistribution doivent être mis en place avant le site pilote. Cependant la télé-installation a des limites : si le volume des composants à installer sur les postes de travail est de plusieurs centaines de Mo, le téléchargement par blocs de 50 Mo prendra plusieurs heures et il sera difficile à maîtriser. Mieux vaut alors procéder par chargement manuel, ce qui suppose un recours à la sous-traitance.

Le parc des postes de travail est souvent composé de postes appartenant à des générations différentes : il est donc hétérogène. Si certains postes ne sont pas éligibles pour la mise en œuvre de l'application (taille de la RAM notamment), il faut les mettre à niveau. Cette intervention sur les matériels impliquera elle aussi souvent le recours à la sous-traitance.

On doit distinguer deux types de sous-traitant : l'installateur et la tierce maintenance. La maintenance du poste de travail doit être mise en place en même temps que l'on déploie l'application. Il faut mettre à la disposition de la tierce maintenance les outils qui permettront d'entretenir le nouveau poste de travail et de former ses personnels.

Le dimensionnement du réseau de l'entreprise est l'une des contraintes que la conception de l'application doit respecter car il conditionne les temps de réponse. Tout se passera bien si le réglage du réseau ne suppose qu'un paramétrage de ses composants, mais on se retrouvera bloqué si l'application pose un problème structurel et non anticipé d'architecture du réseau (il en sera de même si l'application n'a pas anticipé les questions de middleware et de SGBD).

* *

La maîtrise d'ouvrage doit assurer la communication, la formation des utilisateurs et la préparation du support aux utilisateurs. Le plan de communication doit préparer la communication institutionnelle (information du directeur général, des directeurs régionaux) et la communication opérationnelle (information des directeurs régionaux, des directeurs départementaux et des directeurs d'établissement).

La formation vise les utilisateurs ainsi que les tuteurs qui vont former les utilisateurs, mais aussi les techniciens d'installation, de maintenance, de production et de support. Elle obéit à une logistique complexe sous une

contrainte d'échéance très précise car elle doit accompagner de près le déploiement physique de l'application.

Le support est assuré par des centres d'appel dont les personnels doivent être formés à la nouvelle application. Il faut que les opérateurs du centre d'appel disposent d'un guide de questionnement pour outiller leur dialogue avec les utilisateurs. Les statistiques que l'on pourra produire à partir de la base de données du help-desk donneront des indications sur le fonctionnement de l'application, sur les besoins de formation complémentaire - et même sur l'état du marché lorsque l'application traite la relation avec les clients.

Il faut identifier parmi les actions à réaliser sur chaque site celles qui serviront de jalon pour évaluer l'avancement du déploiement. Le tableau de bord du déploiement doit être accessible sur l'Intranet. Il faut le préparer deux à trois mois avant le début du déploiement.

13.4.7 Évaluation

Il est de bonne règle de procéder à une évaluation après la fin d'un projet, lorsque les utilisateurs ont pris le produit en main : cela permet de tirer les leçons de l'exercice et de faire progresser les méthodes en vue des projets futurs. Il est cependant très rare que les entreprises procèdent à de telles évaluations : lorsqu'un projet est terminé, l'attention est attirée par d'autres projets ; si le projet a connu des épisodes pénibles (allongement des délais et du coût, suppression de fonctionnalités), on préfère ne pas « retourner le couteau dans la plaie », ne pas « faire la chasse aux coupables » et on laisse « les cadavres dans les placards ». En cours de route, on avait parfois sanctionné les « porteurs de mauvaises nouvelles » et on pense rarement à les réhabiliter - si même on ne leur fait pas porter la responsabilité de l'échec qu'ils avaient annoncé.

Pour évaluer un projet, il convient de répondre aux trois questions suivantes : le produit fournit-il le service attendu ? le projet s'est-il bien déroulé ? quelles leçons doit-on tirer du déroulement du projet ?

1) *Le produit fournit-il le service attendu ?*

Une première évaluation a été fournie par la recette fonctionnelle ; il faut la compléter, après la mise en œuvre du produit, en s'assurant que le service est efficace en regard des exigences professionnelles. Une enquête auprès des utilisateurs permettra de s'assurer de la pertinence des spécifications et de la bonne insertion du produit dans le système d'information : est-elle « sans couture » ? pose-t-elle problème (doubles saisies, difficultés de codage et de vérification, performances) ?

On pourra ainsi voir des faiblesses que la mise en exploitation révèle mais qui n'avaient pas été perçues lors de la recette ou du site pilote. Si l'on détecte de telles faiblesses, sont-elles tolérables ou faut-il préparer une évolution du produit ?

2) *Le projet s'est-il bien déroulé ?* La réalisation a-t-elle été conforme en termes de qualité (fonctionnalités, performances) ; de délais de livraison ; de coût ? Si des dépassements par rapport aux engagements initiaux se sont

produits, sont-ils acceptables? Peut-on estimer rétrospectivement que l'on aurait dû choisir un autre fournisseur parmi ceux qui avaient répondu à l'appel d'offres? Aurait-il été plus raisonnable d'arrêter le projet en cours de réalisation?

Le coût d'exploitation auquel l'entreprise devra faire face est-il celui qui avait été prévu? Il faut à ce stade réévaluer ce coût d'exploitation, le coût du maintien en condition opérationnelle et le coût de la maintenance évolutive: s'il apparaissait que ces coûts sont trop élevés, une réforme du produit deviendrait nécessaire.

La solution technique mise en œuvre est-elle conforme à l'état de l'art? Si l'on avait à traiter le problème aujourd'hui, choisirait-on la même solution? Au cas où l'on retiendrait une autre solution, quelles seraient les différences en matière de coût de réalisation et de coût d'exploitation? Faut-il envisager de refaire prochainement le produit selon une nouvelle solution moins coûteuse? Quelle est la durée de vie que l'on peut lui attribuer?

Les relations avec le fournisseur ont-elles été correctes? Peut-on envisager de travailler de nouveau avec lui?

Le comportement de l'entreprise a-t-il été correct :

- du côté de la maîtrise d'ouvrage : qualité, stabilité et délai de fourniture des spécifications , réactivité en réponse aux questions de la maîtrise d'œuvre et du fournisseur , qualité de la recette fonctionnelle, qualité du déploiement, de la formation des utilisateurs ;

- du côté de la maîtrise d'œuvre : qualité de l'insertion du produit dans le système d'information, stabilité des spécifications techniques ; réactivité en réponse aux questions du fournisseur ; qualité de la recette technique.

3) *Quelles leçons tirer du déroulement du projet?* En cas de dépassement des coûts et délais : avait-on été trop optimiste dans les évaluations initiales? le fournisseur avait-il pratiqué le *dumping* pour obtenir le marché? la conduite du projet a-t-elle été laxiste?

En cas de carences du côté de l'entreprise : quels sont les points à améliorer dans l'organisation de la conduite de projet? le fonctionnement des comités de pilotage et de suivi? la répartition des responsabilités entre la maîtrise d'ouvrage et la maîtrise d'œuvre?

Quels sont les points à améliorer dans les méthodes utilisées : le reporting, les alarmes, le suivi d'avancement, le suivi des risques? le suivi de l'application des décisions? les procédures d'arbitrage?

Chapitre 14

Vers la maturité

On associe souvent au mot « organisation » le dessin d'un organigramme, arbre renversé qui représente le découpage des responsabilités légitimes et la hiérarchie du commandement. Du directeur général, stratège suprême, un « râteau » descend vers les DGA, puis vers les directeurs. Cependant certaines directions simples ne sont rattachées à aucun DGA et sur le tronc de l'arbre se raccordent quelques branches latérales (figure 14.1) :

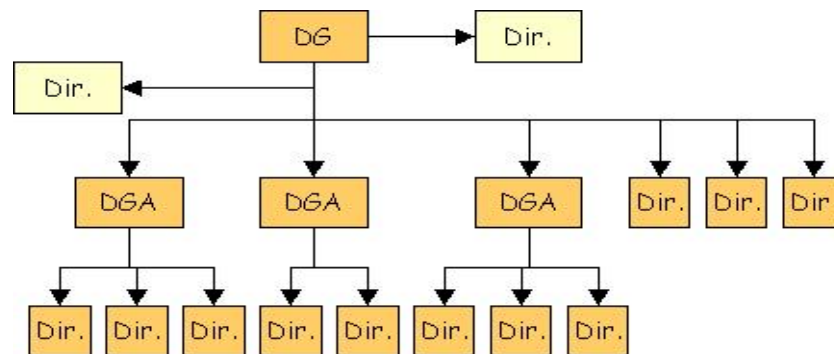


FIG. 14.1 – *L'organigramme de l'entreprise*

L'organigramme délimite les sphères de pouvoir et circuits de communication, balise les sentiers que la carrière peut suivre, définit autant de « villages gaulois » qui servent de repères aux identités. Chargé de tant de symboles, il est naturel qu'il soit fallacieux : il ne faut pas le prendre au pied de la lettre.

Le découpage officiel des directions masque souvent un découpage officieux : l'affichage des missions et responsabilités vise, dans la logique du « compromis managérial » (voir page 566), à éviter des conflits et à calmer les amours-propres. Certains directeurs ont, en fait, plus de pouvoir que les

DGA dont en principe ils dépendent¹. Les directions rattachées directement au directeur général, et placées au dessus du « râteau » des autres directions, sont soit des nœuds de pouvoir important, soit des unités que l'on a mis là parce qu'on ne savait pas où les placer (comme l'inspection générale qui est parfois le « cimetière des éléphants » où l'entreprise case d'anciens directeurs dont l'avancement a été stoppé).

Les rapports entre les directions devraient idéalement être du type coopératif: chacune fait son travail en collaborant loyalement avec les autres, pour le plus grand bien de l'entreprise. On peut estimer cependant que seules 10 % des entreprises fonctionnent de la sorte, sous l'impulsion continue d'un stratège vigilant.

Le mode consensuel, selon lequel toute décision fait l'objet d'un compromis au sein du comité de direction, est beaucoup plus répandu notamment (mais pas seulement) dans les administrations. Il s'impose dans 30 % environ des entreprises. C'est lui qui fait la plus grande part au « compromis managérial ».

Enfin, le mode conflictuel prévaut dans 60 % des entreprises: chaque direction poursuit une stratégie qui lui est propre et entretient avec les autres directions une guerre de frontières, seul l'arbitrage du directeur général pouvant assurer que l'intérêt global de l'entreprise est pris en considération (les pourcentages ci-dessus ne résultent pas d'une enquête, mais d'une concertation entre consultants; et dans une même entreprise peuvent cohabiter, selon des doses variables, les trois modes de relation).

L'interprétation de l'organigramme officiel nécessite donc un commentaire officieux qui, par définition, n'est jamais publié mais qui seul permet de reconstituer l'organigramme « vrai » dont nous donnerons ci-dessous des exemples.

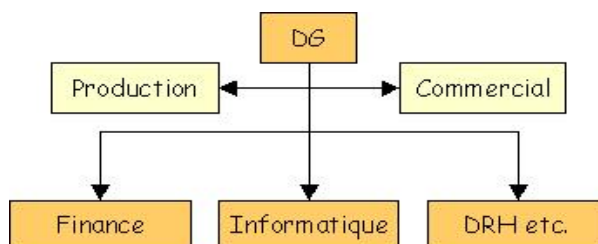
* *

Le découpage des DGA et des directions se cristallise lors de l'histoire de l'entreprise et varie de l'une à l'autre. Cependant toute entreprise produit et vend, et comporte aussi des fonctions « support »: gestion des ressources humaines, finance, informatique, juridique, communication. Parfois, elle fait de la R&D, parfois elle s'est associée à des partenaires.

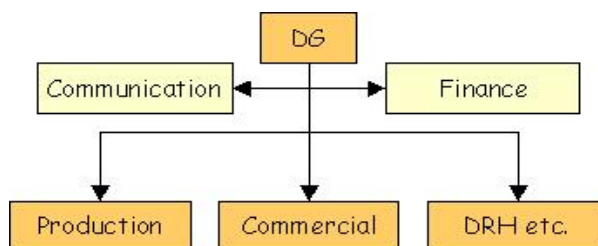
Supposons que nous sachions reconstituer l'organigramme « vrai », et que nous placions tout en haut, comme branches latérales de l'arbre, les directions ou DGA les plus importantes pour le directeur général, celles dont la dialectique anime la vie de l'entreprise et propulse son évolution: l'organigramme ainsi reconstitué révèle les priorités et la culture de l'entreprise.

Lorsque l'entreprise est dominée par ce que nous appellerons la *culture d'ingénieur* (figure 14.2) le directeur général s'intéresse principalement à la production et au commercial et il accorde beaucoup de soin à l'arbitrage entre ces directions dont la dialectique est vive: la production s'étonne des

1. Le « coup du cavalier » est la manœuvre par laquelle un subordonné établit des rapports personnels confiants avec le supérieur de son chef direct, sautant ainsi comme aux échecs par dessus ce dernier. Dès lors celui-ci ne peut pratiquement plus exercer d'autorité sur ce subordonné.

FIG. 14.2 – *Priorité à la production et au commercial*

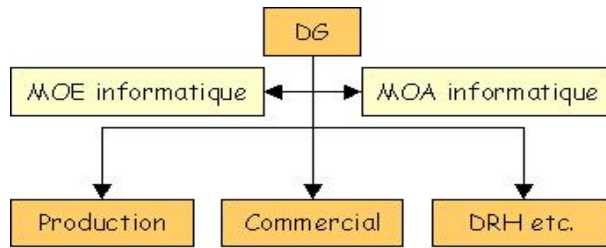
mauvaises performances du commercial, le commercial prétend que les produits sont invendables. Les autres directions sont considérées comme des fonctions support de ces deux activités. Cette forme d'organisation, qui convient bien aux entreprises industrielles, était la plus courante avant les années 1970 (dans les entreprises les plus innovantes, l'articulation principale se serait située entre la R&D et le commercial).

FIG. 14.3 – *Priorité à la finance et à la communication*

Dans le courant des années 1980 et 1990 l'économie s'est « tertiarisée » et on a vu se répandre la *culture médiatique* : le directeur général accorde l'essentiel de son attention à l'image de l'entreprise auprès du conseil d'administration, des créanciers et de la Bourse ; les directions qui accaparent son attention sont alors la communication et la finance (figure 14.3). Les autres fonctions de l'entreprise, y compris la production et le commercial, sont considérées comme des supports à la gestion de l'image.

Supposons enfin que le directeur général prenne au sérieux des phrases comme « le système d'information est stratégique », « l'essentiel de la valeur est créé par le système d'information » etc. Alors il accordera l'essentiel de son attention aux deux pôles responsables du système d'information, confiés chacun à une direction (figure 14.4).

Il convient que la relation entre ces deux directions soit mutuellement respectueuse (il en était de même, dans la phase industrielle, de la relation entre les directions « production » et « commercial » même si elles bataillaient). Des tensions sont inévitables car chacun des deux directeurs a ses propres priorités et doit respecter ses propres contraintes. Il revient donc au directeur général d'assurer une fonction d'arbitre pour éviter l'ex-

FIG. 14.4 – *Priorité au système d'information*

plosion : il doit « faire descendre les barres de graphite dans le réacteur » pour encourager les compromis constructifs.

La dénomination de ces directions n'est pas stabilisée aujourd'hui, et la direction informatique a souvent été baptisée « direction du système d'information » (DSI) ou même « direction des systèmes d'information » au pluriel.

L'administration américaine est en train de mettre en place une organisation de ce type dans ses « agences », qui sont l'équivalent de nos directions d'administration centrale ou de nos ministères. Comme l'appellation « CIO » (*Chief Information Officer*) a été prise depuis longtemps par le responsable de la maîtrise d'œuvre informatique, les directeurs de la maîtrise d'ouvrage prennent le titre de « CTO » (*Chief Technology Officer*). Ces dénominations vont au rebours du bon sens, puisque la maîtrise d'ouvrage et plus proche de l'information et la maîtrise d'œuvre plus proche de la technique ; mais c'est le compromis qui a été trouvé pour partager en deux l'acronyme « IT » (*Information Technology*) sans mettre en question la dénomination déjà attribuée à la maîtrise d'œuvre.

14.1 Professionnalisation de la MOA

Le système d'information, autrefois spécialisé dans quelques fonctions administratives, est désormais présent dans toutes les parties de l'entreprise² tout comme le sont le réseau vasculaire ou le réseau nerveux d'un animal.

Les « concepteurs » de l'entreprise, ces personnes de la direction générale qui assurent la mise en œuvre de la stratégie, le repositionnement sur de nouveaux marchés, l'application de réglementations nouvelles, doivent en tenir compte : la conception du système d'information fait partie de leurs responsabilités. Ils sont, pour une part, « maître d'ouvrage » (délégué ou opérationnel) du système d'information. Que fait un « concepteur » ? il définit le travail que devront faire les agents opérationnels, il rédige les notes techniques et instructions qui leur sont destinées ainsi qu'à leur encadrement ; il retravaille l'organisation de l'entreprise en modifiant les contours

2. En anglais on dirait qu'il est *pervasive*.

des entités qui la composent, leurs missions, leurs moyens, leurs responsabilités, les indicateurs selon lesquels elles sont évaluées. Enfin, il définit les fonctionnalités du système d'information, il le « modélise ».

La modélisation du système d'information est pour les « métiers » une tâche relativement nouvelle ; certes, les entreprises ont déjà un système d'information (de qualité variable) mais jusqu'à la fin des années 1980 il n'a pas été demandé à la maîtrise d'ouvrage de le modéliser. Elle émettait des expressions de besoin le plus souvent vagues que l'informatique devait reformuler de façon assez précise pour pouvoir passer à la réalisation. Parfois la maîtrise d'ouvrage ne se reconnaissait pas dans cette reformulation d'où discussions, conflits et délais ; il arrivait aussi que la maîtrise d'ouvrage changeât d'avis et modifiât sa demande, d'où pour l'informatique obligation de reprendre le travail et bien sûr là encore discussions, conflits et délais.

Ce partage des tâches ayant provoqué l'échec de nombreux projets certaines entreprises ont voulu organiser les maîtrises d'ouvrage, ce qui implique de les doter des compétences, ressources et méthodes leur permettant de maîtriser intellectuellement et pratiquement le système d'information : la modélisation, quand elle est bien faite, favorise l'« alignement stratégique » du système d'information tout en rendant plus claire, plus souple et surtout plus raisonnable la relation entre maîtrise d'ouvrage et maîtrise d'œuvre informatique.

* *

Quand un concepteur assure son rôle de maître d'ouvrage opérationnel, il doit suivre la démarche suivante avant de lancer un projet nouveau :

- rédiger une expression de besoin en langage naturel (voir page 456) : elle doit consacrer l'accord ou le compromis passé entre les directions de l'entreprise, refléter une compréhension partagée de ce que l'on entend faire, et restera la référence principale pendant la réalisation du projet ;
- réaliser l'« étude OFR » (voir page 461) qui fournira les éléments nécessaires à la décision de lancement.

Une fois le projet lancé, d'autres étapes s'enchaînent (modélisation proprement dite, suivi de projet, recette, déploiement etc. ; voir page 453).

Mais pour une grande part le professionnalisme de la maîtrise d'ouvrage réside dans la maîtrise des méthodes à employer pour établir l'expression de besoin et l'étude préalable, dans la clarté d'esprit qui préside aux consultations, expertises et travaux d'écriture, dans la qualité de leur présentation aux dirigeants qui auront à les valider et devront se les approprier.

On découvre souvent, lorsqu'on met en place ces méthodes, que le travail des concepteurs présentait des défauts dont on ne s'était pas avisé :

- 1) Les difficultés de la rédaction de l'expression de besoin montrent que l'on ne se souciait pas, auparavant, de préciser les périmètres de responsabilité et les prérogatives des acteurs concernés par un projet : certes ces questions finissaient par se régler, mais à la longue et à chaud donc souvent plutôt mal alors qu'il est efficace de les traiter à froid, de façon préventive.
- 2) Les réunions se tenaient souvent sans ordre du jour, sans compte rendu, sans enregistrement des décisions ni suivi de leur exécution : ce n'est

plus tolérable quand il faut gérer une lourde logistique de consultations et validations.

3) Les écrits entrelaçaient souvent, dans un même paragraphe, des considérations relatives à l'opportunité, à la faisabilité et aux risques : « il serait souhaitable de faire ceci, on pourrait le faire de telle façon, mais on risque de rencontrer tel problème etc. » Le plan de l'étude préalable exige que l'on traite séparément ces aspects du projet, ce qui facilite leur discussion.

4) L'évaluation de l'opportunité (donc du profit apporté par le projet) est l'affaire de la maîtrise d'ouvrage, l'évaluation de la faisabilité technique (donc du coût du projet) est une contribution de la maîtrise d'œuvre. Si l'étude préalable relève de la responsabilité de la maîtrise d'ouvrage, sa réalisation suppose donc que les deux parties se concertent et se comprennent avant qu'il ne soit question de lancer la réalisation.

Comme ces méthodes contraignent la maîtrise d'ouvrage à réfléchir davantage, son investissement intellectuel est plus précoce ; l'expérience montre que les projets sont ainsi mieux conçus et il peut arriver que sur la durée l'effort total de la maîtrise d'ouvrage en soit diminué. La clarté acquise par le projet, le caractère équilibré des relations avec la maîtrise d'œuvre sont aussi des « plus » que l'entreprise apprécie.

Le sérieux nouveau apporté à la conception du système d'information conduit à anticiper - et sans doute à améliorer - les autres travaux des concepteurs : une étude préalable bien faite comporte une première description, certes à grosses mailles, du travail des personnes et de l'organisation de l'entreprise.

* *

La professionnalisation de la maîtrise d'ouvrage a un coût : il faut former les concepteurs à une méthode qu'ils devront savoir mettre en œuvre sans formalisme superflu. Pour le premier projet ainsi traité ils devront fournir un double effort : acquérir la méthode, puis l'appliquer.

Il peut arriver que certains informaticiens, même s'ils s'étaient plaints de l'amateurisme de la maîtrise d'ouvrage, s'inquiètent d'une évolution qui les « dépossède » d'une partie des tâches qu'ils faisaient auparavant. L'expérience montre que cette inquiétude s'estompe avec le temps mais ses manifestations contribuent aux difficultés de la transition.

L'effort de réflexion que font les concepteurs pour définir le système d'information est bénéfique pour l'ensemble de leur activité : ils deviennent plus précis, ils réfléchissent de façon plus précoce sur les questions d'organisation et sur le travail des personnes. Il peut arriver que le changement d'ambiance qui en résulte déconcerte les personnes qui encadrent ces concepteurs, les maîtres d'ouvrage stratégiques, auxquels il sera demandé par ailleurs de prendre des décisions plus explicites.

Les services où se pratique un *activisme velléitaire*, forme de travail épuisante et peu productive qui submerge les agents opérationnels d'instructions et notes techniques versatiles, sont bousculés par la professionnalisation de la maîtrise d'ouvrage : ils lui seront donc souvent hostiles. Comme tout pro-

professionnel sérieux ressent la satisfaction que procure un travail clairement défini et correctement exécuté, la professionnalisation des maîtrises d'ouvrage finira par s'imposer sauf dans les entreprises malades. Mais l'itinéraire vers cette professionnalisation comporte des obstacles qu'il faut aujourd'hui percevoir, évaluer et contourner.

14.2 Servitudes et grandeur

14.2.1 Le côté de la maîtrise d'ouvrage

Par un abus de langage que nous adopterons ici nous-mêmes, on appelle souvent « maître d'ouvrage » tout court la personne que l'on devrait appeler « maître d'ouvrage délégué ». Placée près d'un dirigeant (président, directeur général, directeur au sein d'une entreprise), elle l'assiste dans la formulation des priorités en matière de système d'information et dans la définition d'un système d'information répondant à la stratégie. L'addition d'obligations contradictoires auxquelles elle est soumise constitue une servitude. Celui qui l'assume lucidement accède à la modeste grandeur qu'a évoquée Alfred de Vigny dans *Servitude et grandeur militaires* (1835).

* *

Toute entreprise est organisée autour de quelques spécialités qu'elle maîtrise et que l'on appelle « métiers ». Il est rare que le système d'information figure parmi ces spécialités. C'est le cas chez American Airlines ou chez FedEx, dont les dirigeants connaissent parfaitement leur système d'information, mais le plus souvent les compétences des métiers s'appuient sur un système d'information mal maîtrisé.

Lorsque le spécialiste du système d'information arrive dans un métier il est d'abord impressionné par la technicité de celui-ci : dans le cockpit d'un avion de ligne, dans un hôpital, dans une salle de marché, dans une usine, la virtuosité des personnes qui utilisent des équipements complexes saute aux yeux. Mais l'examen attentif fait bientôt apparaître des carences. Le système d'information de l'avion de ligne ou du bateau de guerre est « ringard », car la conception de ces machines demande quelques années, après quoi elles ont une durée de vie de quelques dizaines d'années. Le système d'information de l'hôpital, pourtant vital au sens précis du terme, est d'un désordre inouï. La salle de marché, où les écrans abondent, n'est tout comme l'usine pas vraiment bien équipée.

Tout métier, vu de près, est par ailleurs une superposition géologique d'habitudes, vocabulaires, méthodes, logiques, outils définis à des époques diverses et qui délimitent des micro-métiers. Le système d'information reflète et pérennise le désordre sémantique, le fouillis de procédures que chérit leur sociologie. Il en résulte des redondances, délais, surcoûts, erreurs dont les dirigeants s'accommodent avec une bonhomie étonnante. Il faudrait, semble-t-il, que l'entreprise fût menacée dans son existence même pour qu'elle consentît à être tout simplement rigoureuse, logique, claire, méthodique. C'est la lutte pour la survie qui explique que les systèmes d'in-

formation du secteur « banques et assurances » aient cinq à dix ans d'avance sur les autres.

Beaucoup de personnes réduisent l'entreprise aux évidences triviales du « business is business ». Cependant les articulations qu'elle opère entre la pensée et l'action, entre le projet et la réalité, posent des problèmes qui mériteraient l'attention d'un Aristote. Beaucoup de dirigeants croient pouvoir les faire traiter par les seuls informaticiens. Il en résulte une confusion entre les concepts sémantiques des métiers d'une part, les concepts techniques de la plate-forme informatique d'autre part. Nous allons examiner l'une après l'autre les trois dimensions pratiques de la maîtrise d'ouvrage : intellectuelle, organisationnelle et humaine.

Place dans l'organisation

L'insertion du MOAD dans l'organisation dépend de la qualité des relations qu'il entretient avec ses trois interlocuteurs immédiats : le MOAS ; les MOAO ; le responsable de domaine à l'informatique.

Le maître d'ouvrage est ainsi placé au nœud d'un réseau relationnel délicat et certains d'entre eux n'ont pas su l'assumer correctement : ils se sont comportés dans le métier comme une « informatique *bis* » au mauvais sens du terme, imposant aux MOAO des procédures tracassières, allongeant les délais de décision, créant une situation de pouvoir et non une relation de service. Il en est résulté des conflits qui ont conduit à supprimer ces maîtrises d'ouvrage, puis à les créer de nouveau sous un autre nom (le terme « maître d'ouvrage » étant désormais grillé) au prix d'une confusion des missions et des idées.

La coopération avec « l'informatique » est parfois compromise par une disposition bizarre mais fréquente : l'affectation du total du budget des projets à la DSI. Imaginez que vous ayez emprunté à une banque pour faire construire votre maison mais que le financement des travaux ne passe pas entre vos mains, la banque payant directement l'entrepreneur. Si en cours de chantier vous estimez devoir bloquer les paiements, vous devrez passer par la banque et apporter, lors d'un débat contradictoire, la preuve de l'inadéquation des travaux au cahier des charges, de leur mauvaise qualité etc. Dans le rapport entre le client et l'entrepreneur, une telle procédure favorise celui-ci car comme elle contraint le client à expliquer à un tiers pourquoi il n'est pas satisfait elle l'incite à se soumettre aux exigences, aux habitudes, voire aux caprices de l'entrepreneur.

Personne, sans doute, n'accepterait de faire construire une maison dans de telles conditions. Mais c'est souvent ainsi que sont construits les systèmes d'information : lorsque le CSSI a choisi la liste des projets à réaliser, une enveloppe budgétaire est attribuée à l'informatique et elle ne pourra pas être remise en question lors de la réalisation. Les métiers sont alors privés de l'arme la plus naturelle et la plus puissante d'un client : le refus de payer une livraison non conforme.

Certaines entreprises parmi les plus avancées en matière de système d'information ont confié aux maîtrises d'ouvrage la responsabilité du budget informatique (coût des projets, et aussi coût d'exploitation et de maintenance). Elles autorisent par ailleurs les maîtrises d'ouvrage à mettre la DSI en concurrence avec des prestataires extérieurs. L'informatisation du métier est ainsi mise en balance avec d'autres types d'investissement, ce qui incite opportunément les maîtrises d'ouvrage à la sobriété. Mais cela suppose qu'elles possèdent des compétences, une autorité que peu d'entreprises leur attribuent.

Les systèmes d'information des divers métiers sont autant de parties du système d'information de l'entreprise ; certains des éléments de ce dernier doivent être définis au niveau de l'entreprise elle-même, car ils conditionnent l'efficacité d'ensemble en organisant ou facilitant les échanges entre les métiers. S'il est naturel que chaque métier souhaite maîtriser ses propres processus, cela ne doit pas être poussé au point où l'on devrait reconstruire dans chaque métier des éléments qu'il convient de partager avec d'autres. Imaginez une ville où chaque propriétaire voudrait disposer de sa propre source d'eau potable, de sa propre production d'électricité, chaque immeuble ayant quatre murs porteurs pour éviter de s'appuyer sur les immeubles voisins : elle serait coûteuse et peu hygiénique. Il revient aux maîtres d'ouvrage - notamment au MOAD placé auprès du directeur général - d'aider l'entreprise à surmonter le cloisonnement de l'organisation pour y instaurer les processus transverses et les coopérations qui sont l'équivalent des réseaux d'eau, d'électricité, de téléphone, d'assainissement d'une ville.

Il arrive enfin - c'est même banal de le dire - que les progrès du système d'information impliquent de modifier le partage des missions et des responsabilités entre directions. Alors l'organisation résiste. On dit souvent qu'il faut une « conduite du changement » pour faire accepter par les utilisateurs les nouveautés que le système d'information leur apporte, et en effet ils ont souvent besoin d'une formation. Mais dans bien des cas il faudrait plutôt un « changement de la conduite », du mode de direction et d'animation ainsi que de la répartition des pouvoirs entre dirigeants ; l'expérience montre que ce changement-là est le plus difficile à obtenir.

Aspect intellectuel

Les systèmes d'information ambitionnent aujourd'hui non seulement d'automatiser les tâches administratives, mais d'équiper chaque processus de travail au plus près de la pratique opérationnelle. L'automate programmable étant devenu l'assistant de chaque opérateur humain, le « travail assisté par ordinateur » se généralise. Pour définir un système d'information, il faut alors « modéliser le métier ». Le modèle devra être mis en œuvre sur une « plate-forme informatique » mais celle-ci ne résume pas le système d'information, pas plus que le réseau téléphonique ne résume les conversations qui, sans lui, ne pourraient pourtant pas avoir lieu. La plate-forme permet d'incarner la sémantique du système d'information tout comme la physiologie humaine permet d'incarner le langage (idéal) dans la parole (sonore), mais elle ne détermine pas cette sémantique qui doit être définie ailleurs.

Alors que la plate-forme informatique est soumise à des contraintes physiques, techniques et économiques de vitesse, volume et prix, les objectifs fonctionnels du système d'information doivent obéir à des contraintes sémantiques de pertinence, sobriété et cohérence (voir page 255).

La modélisation doit articuler les objectifs fonctionnels aux contraintes de la plate-forme. L'architecture d'un système d'information ressemble à celle d'un immeuble : comme lui, il a des fondations, des piliers et des murs porteurs, des lieux de passage et des espaces privés, des escaliers et des ascenseurs, des réseaux. Ses « fondations » sont à la fois sémantiques (processus, référentiel) et physiques (mémoires, processeurs, langages, interfaces, protocoles et réseaux).

L'amateur exercé qui voit un immeuble peut dans une certaine mesure inférer la solidité de ses fondations en examinant ses formes apparentes ; mais rien ne « saute aux yeux » dans un système d'information. Son exploration à travers l'étroite fenêtre de l'écran ne peut être que progressive. Les contraintes de cohérence et de solidité ne seront donc perçues que par des spécialistes : les utilisateurs subissent les effets des incohérences mais ne peuvent pas deviner leur origine.

En fait la qualité de l'architecture du système d'information a peu de défenseurs. Les dirigeants, qui décident de l'attribution des budgets et du lancement des projets mais ne sont généralement pas eux-mêmes des utilisateurs, se défient de l'« esprit de système » de ceux qui réclament la cohérence. Les informaticiens ont tendance à se laisser accaparer par les grands projets : or un système d'information construit projet par projet sans souci de cohérence finira par ressembler à un de ces immeubles mal bâtis ou mal entretenus où l'on trouve des pièces murées, des couloirs qui ne vont nulle part, des escaliers qui débouchent sur le vide, des réseaux qui s'engorgent et débordent. Il arrive aussi que les informaticiens, soucieux d'abord de qualité technique, accordent peu d'attention à la qualité sémantique : si celle-ci n'est pas défendue par un maître d'ouvrage elle sera négligée.

Le charme que peut avoir la superposition d'architectures diverses dans les quartiers anciens d'une ville vient non pas du désordre, mais des mille dispositions de détail ingénieuses qui ont instauré l'harmonie dans la diversité. Beaucoup de système d'information sont des « bidonvilles de luxe » construits de bric et de broc. La conjonction de la laideur et du gaspillage manifeste à la fois l'arrogance de la richesse et la carence de la pensée, seul celui qui sait voir l'architecture d'un système d'information peut ressentir la répugnance qu'inspire un tel spectacle.

Questions humaines

Alors que les défauts du système d'information sautent aux yeux du maître d'ouvrage il est souvent le seul qui puisse les diagnostiquer. Les utilisateurs, s'ils pestent contre « l'informatique », sont loin de percevoir l'origine des difficultés ; elle se trouve d'ailleurs plus souvent dans leur métier que chez les informaticiens. Les dirigeants des métiers, pour leur part, ont coutume de qualifier le système d'information de « stratégique » mais ignorent que la stratégie en matière de système d'information se construit

par de patientes, de persévérantes dispositions de détail. Ce que le maître d'ouvrage propose pour progresser dans la pertinence, la sobriété, la cohérence, leur paraît mesquin et terre-à-terre en regard de l'idée emphatique qu'ils se font de la Stratégie, et d'autant moins convaincant que cela risque de soulever des problèmes de personne : « pas de vagues » est la première règle de beaucoup de dirigeants.

Le maître d'ouvrage n'a d'ailleurs auprès d'eux qu'un pouvoir d'influence, de conseil. Il ne maîtrise ni les décisions ni les budgets : il est le « conseiller du prince ». C'est un rôle visible, glorieux et envié, car dans l'entreprise peu de personnes ont la possibilité de parler au président, au directeur général, aux DGA ou aux directeurs, mais ce rôle a deux faces. Si la face visible est brillante, le conseiller du prince expérimente chaque jour l'autre face, terne et frustrante : ses avis ne sont pas souvent écoutés ; il doit attendre longtemps l'occasion propice pour faire passer une idée de simple bon sens ; il se fait parfois désavouer ; il arrive qu'il se croie, se sente ou se sache méprisé ; on lui fera, à l'occasion, porter la responsabilité de décisions contraires à ses recommandations.

Témoin de décisions malencontreuses et de budgets jetés par la fenêtre, il est souvent écrasé par son impuissance devant des problèmes qui lui crèvent les yeux. Sans doute il ne reste pas muet, mais l'entreprise est sourde. Il voit la perspective reculer à mesure que le temps passe.

Pourtant l'entreprise bouge, même si c'est lentement et avec retard. S'il se retourne vers le passé et regarde le chemin parcouru, le maître d'ouvrage voit, non sans surprise, qu'il est résulté quelque chose de l'addition de tant de journées apparemment stériles. Pas à pas, une distance a été parcourue. La vague a usé la falaise. Il a fallu pour cela se tenir à l'affût, attendre l'occasion pour obtenir la décision judicieuse. Mais rester vigilant à l'affût est un art. Parfois, à force d'attendre, le chasseur s'endort et quand la chance se présente, il la laisse passer parce qu'il ne l'a pas vue.

Il peut aussi finir par radoter. Vous savez par exemple que l'entreprise devrait améliorer son référentiel et mettre en place à cette fin une administration des données, maîtriser la qualité de ses procédures en implantant quelques workflows, améliorer la définition du poste de travail etc. Vous le dites mais rien ne bouge. Vous le répétez encore et encore : cela vous transforme en perroquet. Vous devenez dogmatique, schématique. N'ayant pas, et pour cause, entamé la mise en œuvre, vous êtes privé des enseignements de l'expérience. Il y a du vrai dans ce que vous dites, mais vous êtes devenu incapable de discerner les difficultés pratiques et cela se sent. Vous êtes comme ces officiers du *Désert des Tartares* de Dino Buzzati, qui vieillissent et se disqualifient dans l'attente de la bataille.

Pour éviter ce naufrage, il faut se consacrer à des réalisations partielles. Vous n'avez pas pu faire passer la « grande idée », si vraie, si juste, qui vous tient tant à cœur ? Faites donc avancer de petites affaires. Restez vigilant à l'affût de la grosse bête mais pour meubler l'attente braconnez le petit gibier. Soyez actif et manœuvrier. Formez des compétences. Affûtez les méthodes. Tuez dans l'œuf les projets inutiles. Préparez, pour le dirigeant que vous assistez, une formulation claire des choix et priorités : même s'il ne vous suit

pas il en restera quelque chose. Constituez un réseau d'amis, de complices avec lesquels vous pourrez partager vos soucis, vos colères et votre analyse, et ourdir la manœuvre qui orientera l'entreprise vers la décision.

Si, en effet, le système d'information doit se construire dans la perspective d'une stratégie claire, cohérente, cette stratégie doit le plus souvent rester implicite : le processus de décision met en jeu trop de symboles pour être entièrement rationnel, qu'il s'agisse du positionnement de l'entreprise, de l'image qu'elle se fait d'elle-même ou des sphères de pouvoir que découpe son organisation. Vous êtes placé à la charnière entre le monde de l'ingénieur et le monde du stratège, à la charnière de l'expertise et de la décision ; si vous parvenez à assumer ce rôle pénible, si vous faites mûrir votre compréhension de ces deux mondes et de leur articulation, vous serez prêt à assumer de façon convenable la fonction de stratège le jour où cela vous sera demandé.

Respecter les personnes qui travaillent dans l'entreprise, respecter les dirigeants, impose de ne pas les considérer comme des êtres parfaitement rationnels : ce sont des êtres humains de bonne volonté sans doute mais qui ont leurs limites, leurs habitudes, leurs passions. Devant les pathologies de l'entreprise le maître d'ouvrage est comme un médecin. Il doit, en usant de méthodes thérapeutiques semblables à celles de l'École de Palo Alto (Marc et Picard [160]), mettre au service de la stratégie une tactique opportuniste.

* *

À l'issue d'une conférence où j'avais exposé ce qui précède un participant (lui-même maître d'ouvrage dans une grande entreprise) s'écria « s'il en est ainsi, vivement la retraite ! » Il était de ceux qui ont besoin de croire en la rationalité de l'entreprise et qui veulent ignorer qu'étant un organisme vivant elle a, tout comme une personne, besoin de soins médicaux - surtout en ce qui concerne son système d'information, qui est pour elle aussi difficile à comprendre que ne l'est son propre cerveau pour un être humain.

La servitude et la grandeur de la maîtrise d'ouvrage ont attiré vers cette activité des personnes d'un profil particulier. Elles entendent aider l'entreprise à articuler la puissance de l'automate programmable et la compétence de l'être humain. Elles assument pour cela le discrédit attaché à ceux qui s'écartent des carrières habituelles. Elles assument aussi les risques que prend quiconque contribue à la décision sans maîtriser les budgets et sans occuper une position de force dans l'organigramme. Elles postulent - avec un optimisme souvent contredit, toujours renouvelé - que la simple raison, la modeste rigueur s'imposeront à la longue par le poids de leur évidence. L'expérience leur apprend à gérer la frustration, les contrariétés que comporte leur activité quotidienne, à ne pas trop s'émouvoir de la perversité des « hommes de pouvoir », à se réjouir de réalisations partielles, plus raisonnables que rationnelles, et à tenir le cap dans la durée. C'est ainsi en effet que se construisent, dans un lent accouchement, les systèmes d'information de nos entreprises.

14.2.2 Le côté du DSI

On a pris l'habitude, en France, d'appeler « DSI » le directeur chargé de la plate-forme informatique et télécoms. Cet acronyme se déploie en « directeur du système d'information » et parfois - comme si une même entreprise pouvait avoir plusieurs systèmes d'information - en « directeur des systèmes d'information » au pluriel. Aux États-Unis, on le nomme « CIO » (« Chief Information Officer »).

Il est malencontreux que l'on ait donné un tel titre au responsable de la maîtrise d'œuvre du système d'information : celui-ci n'est-il pas également l'affaire des métiers utilisateurs, ces maîtrises d'ouvrage qui doivent le modéliser, le « spécifier », le définir de façon pertinente en regard de leurs besoins ? Les maîtrises d'ouvrage ne sont-elles pas, à l'intérieur de l'entreprise, les « clients » de la direction informatique ? Le système d'information ne relève-t-il pas ainsi de la responsabilité conjointe de ses maîtrises d'ouvrage et de sa maîtrise d'œuvre ?

Mais laissons là les questions de terminologie. Le DSI, puisque DSI il y a, a toujours été placé sur une crête entre deux versants : le versant « métier » des services rendus aux utilisateurs ; le versant « technique » de la plate-forme informatique et télécoms. C'est une position intéressante mais d'autant plus inconfortable que les versants sont plus abrupts et plus profonds.

Or le versant technique se complique avec l'introduction de la programmation par objets et la cohabitation entre informatique de calcul et informatique de communication (messagerie, Intranet, Web etc.) ; le métier informatique se diversifie en spécialités comme le fit naguère la médecine (middleware, sécurité, réseaux etc.) ; les frontières entre l'externe et l'interne, entre le progiciel et le spécifique, posent des problèmes de positionnement dont les paramètres évoluent.

Le versant métier s'est lui aussi enrichi. Le système d'information, concentré naguère dans de grandes applications, irrigue désormais tous les processus ; le travail assisté par ordinateur se généralise, articulant la puissance de l'automate au jugement de l'être humain. La modélisation nécessite l'apport de l'expertise des métiers et la validation par les dirigeants.

Le DSI peut-il se maintenir en équilibre sur une telle crête ? On peut en douter. La plate-forme informatique doit fonctionner en continu, sans défaillance, tout en absorbant un flux de nouveaux équipements et de nouveaux logiciels. Elle pose des problèmes de maîtrise physique et sociologique des plus délicats. Est-il humainement possible que le responsable d'une telle usine puisse aussi assurer convenablement, et dans la durée, la pertinence, la sobriété, la cohérence sémantique du système d'information, son alignement stratégique etc. ? Ce serait un miracle.

Il faudra distinguer deux fonctions, celle du DSI et celle du responsable des maîtrises d'ouvrage. Le DSI assurera la performance et la qualité de la plate-forme technique tout en la maintenant conforme à l'état de l'art.

Cette séparation des fonctions sera refusé par certains DSI : y voyant une perte de prestige et de pouvoir ils préféreront se tenir à cloche-pied

sur leur crête, même si c'est inconfortable et inefficace. Tel est le poids des symboles ! L'évolution sera donc lente. C'est la rançon du succès de l'informatique : si celle-ci était restée confinée à quelques grandes applications, les informaticiens auraient pu continuer à modéliser le système d'information à partir d'expressions de besoin plus ou moins précises. Mais comme elle est devenue à la fois le langage de l'entreprise et son système nerveux, son modèle ne pourra être pertinent que s'il est établi sous la responsabilité des métiers utilisateurs eux-mêmes, dotés de modélisateurs qualifiés qui seront pour l'informatique des « clients compétents ».

La situation du DSI est aujourd'hui en crise : sa « durée de vie » diminue. Au milieu des années 1990, un DSI restait en fonction 4,7 ans en moyenne ; en 2005, cette durée se réduit à deux ans. Le « turn over » est donc très rapide. C'est que l'entreprise n'est pas certaine d'avoir, demain, besoin d'un DSI alors qu'il est (semble-t-il) si simple d'externaliser l'informatique ou de recourir à des progiciels. La fonction informatique se serait-elle banalisée ?

La « simplicité de l'informatique » est une de ces illusions qui fleurissent lors des périodes de transition, quand un changement de repères suscite le désarroi. Le rôle de l'informatique s'est transformé à la fin des années 1990, avec l'introduction du traitement du langage naturel dans l'informatique de communication, la fusion de la donnée et du commentaire dans les documents XML, l'approche du système d'information par les processus etc. Les outils se sont diversifiés, le discours commercial est devenu de plus en plus séduisant (mais sans doute pas plus sincère).

Les directions générales, qui n'ont jamais été un repaire d'experts en système d'information, sont dépassées. Elles trouvent donc reposant de croire que se débarrasser de l'informatique en l'« outsourcing » auprès d'un fournisseur externe peut résoudre tous les problèmes (Strassmann [195]).

Il est vrai que l'outsourcing résout quelques problèmes. La direction informatique, tout comme l'entreprise dans son ensemble, doit savoir modifier son périmètre. Quelles sont les tâches qu'il convient de réaliser en interne ? Celles qu'il vaut mieux confier à un fournisseur ? Quelles sont, parmi les spécialités selon lesquelles s'est diversifiée la compétence de l'informaticien, celles dont on pourra rentabiliser la formation et l'entretien au sein de la DSI, et celles que l'on devra se procurer auprès des SSII, mieux placées pour les rentabiliser ?

Faut-il conserver quelques programmeurs en interne, ou transformer la direction des études en gestionnaire de contrats ? Parmi les logiciels dont l'entreprise a besoin, quels sont ceux qu'il vaut mieux acheter sur étagère, sous forme de « progiciels » ou d'ERP, quels sont ceux qu'il est préférable de spécifier et de réaliser soi-même (ou de faire réaliser sur mesure par des fournisseurs) ? Lorsqu'un produit est développé par une SSII, comment se l'approprier, comment s'approprier les outils qui ont servi à l'élaborer ?

Ces questions préoccupent le DSI. Il doit donc assurer une veille technologique. Comment évolue l'offre des SGBD ? des EAI ? des ERP ? des machines, réseaux, langages ? Que penser (et faire) de Linux ? de XML ? des Web Services ? de XQuery ? Dans sa relation avec une innovation, le DSI passe par des étapes. D'abord il l'ignore parce qu'elle est née loin de lui. Puis

Un DSI de mes amis m'a donné la recette pour durer trois ans et ainsi accroître de 50 % la durée de ses fonctions par rapport à la moyenne. C'est une caricature mais elle est assez ressemblante pour être instructive. Je la cite donc *in extenso* :

« Tu suggères à l'entreprise d'outsourcer l'informatique. Les fournisseurs soutiennent ta suggestion avec enthousiasme et la font miroiter aux yeux du directeur général.

« Pendant un an, tu fais des études, un appel d'offres, tu compares les propositions, tu négocies avec les syndicats, tu rends compte au comité de direction, ça t'occupe et ça te donne de l'importance. Enfin tu choisis un fournisseur.

« Pendant la autre année tu assures la transition : tu fais passer les données, les applications, le personnel chez le fournisseur tout en maintenant l'informatique en marche. C'est du boulot ! Tu es encore plus occupé.

« Durant la troisième année, l'informatique est enfin outsourcée. On atteint le régime de croisière, mais le fournisseur ne tient pas ses promesses car, bien sûr, il était impossible de les tenir. C'est le désenchantement, un contentieux se crée. Tu consultes des juristes, tu négocies, ça t'occupe encore. Finalement tu fais part au directeur général de ton impuissance à régler les problèmes ; il s'irrite de plus en plus et finit par te virer mais tu auras tenu trois ans, sinon plus.

« Ton successeur dira que l'outsourcing était une erreur et qu'il faut tout ramener dans l'entreprise. Il lui faut un an pour dénouer le contrat et définir la manœuvre, un an (au moins) pour exécuter celle-ci, un an pour reconnaître que ça ne va pas mieux, ce qui entraînera son éviction.

« Le successeur de ton successeur pourra, lui, recommander de nouveau l'outsourcing. Si le directeur général a changé (en six ans, c'est possible), l'entreprise n'aura pas gardé un souvenir précis de sa première aventure et ce DSI sera écouté, ce qui nous ramène au premier paragraphe.

« Une carrière de DSI peut donc se conduire ainsi : quand on est recruté, recommander l'outsourcing si l'entreprise gère elle-même l'informatique et recommander de tout ramener dans l'entreprise si elle a déjà outsourcé. Le scénario se décline en variantes selon que l'entreprise a outsourcé tout ou partie de l'informatique, qu'elle utilise un ERP ou non etc.

« Dans tous les cas cette tactique te donne au moins trois ans ; après quoi tu pourras la rejouer dans une autre entreprise, si tu trouves un autre poste de DSI. »

elle est portée à sa connaissance, les commerciaux étant à l'informatique ce que les visiteurs médicaux sont à la médecine : une alarme s'allume alors dans son esprit, mais il reste en observation. Puis il se décide à lancer une expérience à petite échelle pour voir comment cela marche et faire acquérir par ses collaborateurs un premier savoir-faire. Ensuite il étudie la possibilité de l'utiliser : il en parle au directeur général, la présente au CSSI, prépare l'investissement, forme ses personnels. Enfin, l'innovation est introduite « en vraie grandeur » dans l'entreprise. Parfois l'ensemble de cette démarche est rapide, parfois il prend des années.

Comment faire pour ne pas prendre trop de risques tout en évitant de se mettre en retard par rapport à l'état de l'art ? Comment faire pour ne pas être dupe d'un discours commercial séduisant ?

* *

Tout en réfléchissant à ses frontières et en observant l'évolution technique, le DSI doit veiller au bon fonctionnement de l'informatique : il est responsable d'une usine qui ne doit jamais s'arrêter. Peu importe qu'un PC se « plante » ici ou là, mais une panne générale serait une catastrophe : les chaînes de production s'arrêtent, les files d'attente s'allongent, les agents protestent tout en tentant de se débrouiller, les clients s'énervent, l'image de l'entreprise souffre. L'architecture des serveurs, routeurs, réseaux, doit donc être sécurisée, robuste, supervisée en continu, et les informaticiens doivent être insérés dans une organisation qui garantit la qualité de leurs travaux.

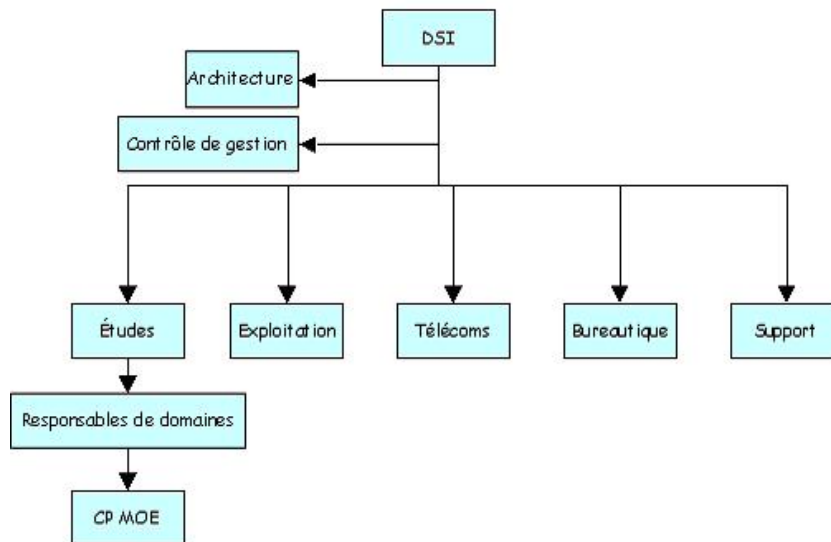


FIG. 14.5 – Organigramme d'une DSI

Or cette ressource humaine est difficile à gérer (figure 14.5) : il s'agit d'une population de spécialistes et les spécialistes seront toujours tentés de s'organiser en corporations mutuellement hostiles. On rencontre à l'intérieur

d'une DSI plusieurs sociologies ombrageuses : celle des opérateurs qui font tourner les serveurs ; des programmeurs qui écrivent les codes et des chefs de projet qui pilotent les contrats avec les SSII ; du support aux utilisateurs, dispersé sur le territoire et dans des centres d'appel ; des administrateurs, des superviseurs ; des hommes des réseaux télécoms ; des responsables de la qualité, de la sécurité, des méthodes, de l'architecture etc. Le DSI doit les recruter, les organiser, arbitrer leurs conflits, faire prévaloir l'intérêt de l'entreprise sur celui de leurs corporations.

* *

L'architecture informatique, c'est un ensemble de « solutions » qui chacune combinent des machines, réseaux, systèmes d'exploitation, logiciels et interfaces. Le choix de chaque solution suppose d'évaluer, au milieu d'un charivari commercial assourdissant, la qualité des « briques » qui la composent, leur aptitude à s'intégrer, la réalité des performances, la pérennité des fournisseurs, la viabilité des techniques.

Comme chaque fournisseur a dans son catalogue quelques mauvais produits qu'il doit pourtant placer, et dans ses équipes quelques mauvais ingénieurs qu'il doit pourtant caser, tout client incompetent sera inévitablement mal servi. Pour obtenir un service convenable le DSI doit donc connaître les fournisseurs, comprendre les contraintes auxquelles ils sont soumis, savoir parler leur langage et se faire respecter.

Le DSI est lui-même un fournisseur pour les métiers de l'entreprise, pour les maîtrises d'ouvrage. Il doit percevoir leurs besoins réels à travers des demandes souvent non priorisées, confuses, inflationnistes et versatiles. Sa tâche sera facilitée s'il a en face de lui une maîtrise d'ouvrage professionnelle, capable d'exprimer des besoins pertinents, sobres, cohérents, stables, de modéliser les processus, fournir des spécifications claires, suivre les projets, former les utilisateurs - bref d'utiliser au mieux les ressources que l'informatique apporte.

Il est vrai que la cohabitation avec une maîtrise d'ouvrage professionnelle suppose une négociation d'égal à égal. Cela sera parfois difficile pour le DSI car, pour se soulager des préoccupations que nous venons de décrire, il peut être tenté par l'ivresse du pouvoir. Quiconque dispose d'un budget annuel de quelques dizaines ou centaines de millions d'euros, et dont une bonne partie est consacrée à des marchés, est la cible des flatteries que prodiguent les fournisseurs (voir page 569). Soumis à l'alternance de cette douche tiède et de la douche froide qui leur est administrée en comité de direction, certains DSI deviennent susceptibles et irritables : le mauvais caractère est chez eux une maladie professionnelle.

Les responsabilités que nous venons de décrire, sont-elles vides, négligeables ? Non, et les entreprises qui croient se débarrasser des difficultés de l'informatique en faisant sauter leur DSI se font des illusions :

- lorsqu'elles signent un contrat d'« outsourcing » le fournisseur est tout sourire : elles découvriront bientôt que la relation avec lui est plus difficile qu'avec une DSI interne, car il est moins proche et se retranche derrière le contrat en utilisant les ressources de la procédure judiciaire ;

- lorsqu'elles accélèrent le « turn over » des DSI, elles empêchent la capitalisation de l'expertise et dégradent le climat chez les informaticiens ;
- lorsqu'elles se jettent à corps perdu dans les bras d'un ERP (et, pourquoi pas, d'un EAI aussi pour faire bonne mesure), elles s'engagent dans une démarche plus complexe qu'elles ne le pensent et dont elles auront tôt fait de découvrir les coûts cachés.

Est-ce à dire qu'il ne faut rien « outsourcer », qu'il ne faut jamais recourir aux ERP et EAI? Certes non! Mais il faut que quelqu'un sache définir avec justesse la frontière de ces prestations, les articuler dans l'architecture de l'entreprise, négocier avec les fournisseurs et les utilisateurs, éclairer la perspective sur les quelques années qui viennent, encadrer la population intéressante mais difficile des informaticiens : et tout cela, c'est la tâche du DSI.

Comment l'entreprise pourrait-elle se passer de lui?

14.3 Check-list du système d'information

Lorsqu'on doit dépanner un moteur, on suppose d'abord qu'il a une des pannes les plus fréquentes puis on va progressivement vers les plus rares : d'abord l'allumage ; puis le filtre à air ; ensuite l'alimentation en essence, le carburateur etc. Cela permet d'expliquer 80 à 90 % des pannes, les autres relevant d'une expertise plus élaborée.

Une *check-list du système d'information* doit permettre d'évaluer la qualité d'un système existant et de prescrire les mesures qui, dans les cas les plus fréquents, permettront de le dépanner au plus vite. La check-list que j'utilise lorsqu'un directeur général demande mon avis commence par l'examen du poste de travail des agents opérationnels ; puis, faisant un saut à l'autre bout du système d'information, elle considère le tableau de bord des dirigeants et des *managers* opérationnels.

Ayant pris ainsi le système d'information en tenaille, on passe ensuite à son architecture (organisation des responsabilités, sémantique) et enfin à sa maîtrise du point de vue de l'évolution fonctionnelle, de la plate-forme technique et de l'économie. La liste de questions ci-dessous est indicative et sommaire : si l'on repère un défaut, il faudra poser d'autres questions pour préciser le diagnostic.

À l'issue de cet examen on peut fournir une première évaluation et formuler quelques recommandations. Si les réponses à la check-list vont toutes dans le bon sens, le système d'information de l'entreprise semble *a priori* robuste : il peut encore présenter des défauts que la check-list n'a pas permis de voir, mais ceux-ci ne sont en tout cas guère apparus à ses utilisateurs.

1) Poste de travail

Les questions porteront d'abord sur le système d'information opérationnel, puis sur l'informatique de communication.

Les agents opérationnels doivent-ils faire souvent des ressaisies manuelles? Doivent-ils, lors d'une même opération, se connecter et se déconnecter à diverses applications? Leurs habilitations sont-elles clairement définies? Les

droits d'accès sont-ils attribués automatiquement lorsque l'agent s'identifie et authentifie son identification? L'agent doit-il s'identifier plusieurs fois par jour?

La gestion des impressions est-elle efficace? Les courriers envoyés par l'entreprise sont-ils de bonne qualité?

Les agents sont-ils convenablement aidés par le système d'information dans l'exécution de leurs tâches?

L'entreprise réalise-t-elle une enquête périodique sur la satisfaction des utilisateurs du système d'information? Des suites sont-elles données aux résultats de cette enquête?

Les utilisateurs reçoivent-ils une formation peu de temps avant la mise en place d'une nouvelle application? Des « piqûres de rappel » périodiques sont-elles prévues?

Les performances de l'informatique, telles qu'elles sont ressenties par les agents devant leur poste de travail (délais d'affichage des écrans, de traitement des données), sont-elles convenables? Les pannes informatiques sont-elles fréquentes? Gênantes?

Le support aux utilisateurs (aide de premier niveau, dépannage, help desk) est-il rapide? compétent? efficace?

L'entreprise a-t-elle déployé une messagerie? Cette messagerie équipe-t-elle *tous* les agents? Ont-ils chacun une adresse personnelle? Est-elle animée (surveillance des bons usages, publicité des bonnes pratiques)?

L'entreprise a-t-elle mis en place une documentation électronique, un Intranet? L'utilise-t-elle pour sa documentation technique et professionnelle? Pour la communication interne? Les documents sont-ils clairement rédigés? Convenablement tenus à jour?

L'entreprise utilise-t-elle les dispositifs de rédaction coopérative, de dissémination sélective?

L'entreprise a-t-elle équipé ses processus administratifs de workflows (demandes de congé et de mutations, approbation des contrats, commandes de matériel et de travaux etc.)? Les indicateurs de qualité sont-ils disponibles? La traçabilité des affaires est-elle assurée?

Les phrases que l'on entend prononcer par les utilisateurs à propos du système d'information sont-elles du genre « l'informatique est encore en panne », « je ne sais pas me servir de cette application », « ça change tout le temps, c'est trop compliqué », « la direction générale fait n'importe quoi » etc., ou du genre: « on sait ce qu'on a à faire », « on est bien outillés », « c'est bien organisé », « la boîte est efficace »?

2) *Système d'aide à la décision*

Les dirigeants disposent-ils d'un tableau de bord de bonne qualité (soberement sélectif, commenté)? Le comité de direction passe-t-il du temps à confronter des statistiques incohérentes?

Le système d'information fournit-il aux *managers* opérationnels les indicateurs qui permettent le suivi quotidien ou hebdomadaire des performances et de la charge de travail de leurs unités?

3) *Architecture*

L'entreprise a-t-elle organisé des maîtrises d'ouvrage déléguées dans ses divers métiers et auprès du directeur général? Ces maîtres d'ouvrage délégués connaissent-ils les besoins des utilisateurs du métier? Ont-ils procédé à une segmentation de cette population? Font-ils une veille sur la façon dont les entreprises analogues utilisent leur système d'information?

Existe-t-il un comité des systèmes d'information? Les méthodes à utiliser pour établir le programme des travaux d'une année, pour assurer l'exécution d'un projet sont-elles définies? Connues?

Les budgets informatiques sont-ils gérés par les maîtres d'ouvrage ou par la direction informatique? La relation entre maîtrise d'ouvrage et maîtrise d'œuvre est-elle convenable, tant en ce qui concerne les relations entre personnes que l'efficacité de leur coopération?

Les nomenclatures (produits, clients, organisation etc.) et les identifiants sont-ils documentés? Résident-ils dans un référentiel consultable dans l'entreprise? L'entreprise dispose-t-elle d'un annuaire des agents? Cet annuaire est-il utilisé pour gérer les habilitations? Les mises à jour des données de référence sont-elles répercutées immédiatement dans les applications?

4) *Maîtrise*

L'entreprise a-t-elle établi un plan d'urbanisme du système d'information? Ce plan d'urbanisme a-t-il fait l'objet d'une communication? Peut-on dire que le plan d'urbanisme a été « approprié » par les dirigeants? par les utilisateurs? Le plan d'urbanisme est-il mis à jour chaque année? Éclaire-t-il les discussions budgétaires?

Le « portefeuille système d'information » de chaque domaine de l'entreprise a-t-il été défini? Son évolution est-elle gérée de façon à assurer l'évolutivité du système d'information?

Les échanges d'information entre domaines sont-ils connus? Gérés? La façon dont les divers domaines utilisent les référentiels de l'entreprise est-elle cohérente?

La direction de l'informatique dispose-t-elle des compétences dont elle a besoin? Utilise-t-elle bien les connaissances informatiques de ses personnels? La mise en œuvre du produit d'un fournisseur est-elle accompagnée par les informaticiens d'efforts de formation et d'appropriation?

La veille technologique permet-elle de connaître l'offre de progiciels? de définir la frontière entre ce que l'on doit confier aux progiciels, et ce qu'il vaut mieux réaliser en spécifique? de définir la frontière entre ce qu'il convient d'externaliser, et ce qu'il vaut mieux réaliser en interne? Est-elle prolongée par des essais à petite échelle (Linux, Web Services etc.?)

Le réseau télécoms est-il bien dimensionné pour les besoins présents? Les besoins futurs? la part des dépenses télécoms dans le budget informatique est-elle excessive, raisonnable? les solutions télécoms sont-elles conformes à l'état de l'art?

L'architecture informatique (middleware etc.) est-elle conforme à l'état de l'art? Est-elle convenablement sécurisée (sauvegarde, pare-feux etc.)? Les travaux de réfection nécessaires ont-ils été engagés? Les dépenses de

maintenance sont-elles dans un rapport convenable avec les coûts prévisibles de réfection? Les solutions peu coûteuses (groupware, Intranet, workflows, Web etc.) sont-elles utilisées volontiers par l'informatique? Possède-t-elle les compétences nécessaires pour les concevoir et les exploiter?

La fonction de coût du système d'information est-elle connue? Les classifications comptables sont-elles correctes (séparation entre dépense d'exploitation et investissement; connaissance des coûts de la maîtrise d'ouvrage; connaissance des coûts internes)?

La dynamique du coût du système d'information (développement, maintenance) est-elle maîtrisée? Peut-on estimer que le degré d'informatisation de l'entreprise est convenable? Est-elle sous-informatisée, sur-informatisée?

14.4 Un système d'information mûr

Pourquoi les systèmes d'information de nos entreprises ne sont-ils pas mûrs? Parce que souvent elles distinguent mal le gros œuvre de la finition. Le gros œuvre fonde la solidité du système (par analogie avec un immeuble : fondations et murs porteurs solides, dimensionnement des accès aux réseaux), la finition permet de rendre des services utiles (les fenêtres ferment bien, les peintures sont de bonne qualité, les tapisseries sont jolies). Si le gros œuvre est mal réalisé, le système d'information reste fragile et incohérent; ce n'est donc pas un système mais un *machin*, même s'il présente des aspects séduisants.

Certaines entreprises s'intéressent davantage aux projets en cours, à leurs épisodes glorieux ou conflictuels, aux « paillettes », qu'à la solidité du gros œuvre dont les avantages ne peuvent apparaître qu'à moyen terme. Celui qui entend s'occuper du gros œuvre s'attirant alors des ennuis, les vocations sont rares.

Les entreprises parvenues les premières à la maturité du système d'information bénéficient d'un avantage concurrentiel : plus productives, plus souples, elles peuvent faire évoluer plus facilement leur positionnement et tirer parti des partenariats. Cela leur procure un profit supérieur à la norme que la concurrence instaure dans leur secteur d'activité.

Lorsque les entreprises auront toutes mûri leur système d'information, qu'elles maîtriseront toutes la façon dont il concrétise et outille leur langage et leurs processus de travail, lorsqu'elles seront toutes devenues évolutives et souples manœuvrières, le système d'information sera devenu une composante du capital productif semblable aux autres. L'avantage concurrentiel qu'il apporte disparaîtra; ce sont les consommateurs qui, en définitive, bénéficieront du gain d'efficacité sous la forme d'une hausse du rapport qualité/prix.

Mais ce n'est pas pour tout de suite. La maturation de l'entreprise autour du système d'information sera la grande affaire du XXI^e siècle et peut-être encore des siècles suivants. La modification des valeurs, priorités et organisations qu'elle implique demandera longtemps, de même qu'il a fallu longtemps pour que les entreprises sachent utiliser l'énergie d'origine fossile et

l'électricité³. Les difficultés que nous rencontrons aujourd'hui, les obstacles étonnants qui s'opposent à des mesures peu coûteuses et de simple bon sens comme la mise en place des workflows donnent la mesure des changements que le système d'information provoque ou nécessite.

C'est pourquoi il est intéressant d'observer dès aujourd'hui les systèmes d'information mûrs, qui ne le sont d'ailleurs souvent qu'en partie : les plus avancés permettent de se représenter dès aujourd'hui, d'anticiper ce que seront les entreprises et l'économie de demain.

* *

Le système d'information de certaines entreprises est enfin arrivé à maturité. Pour cela, il a fallu se battre, surmonter des obstacles, modifier l'organisation, faire accepter de nouvelles valeurs et de nouvelles compétences. La lutte a été longue mais elle a porté ses fruits : le référentiel est en bon état, les processus de production sont bien outillés, les composants⁴ sont modulaires (ce qui facilite leur évolution), le « bus » qui les relie au sein du système informatique⁵ est bien maîtrisé, l'interopérabilité avec les partenaires fonctionne bien⁶, des workflows équipent les procédures administratives dont ils permettent de contrôler la qualité, les divers médias sont convenablement articulés⁷, le système d'aide à la décision est en place, l'Intranet et l'Internet sont entrés dans les mœurs.

Le système d'information a ainsi été *urbanisé*. Les utilisateurs sont satisfaits ; ils trouvent d'ailleurs tout cela aussi simple, aussi naturel que l'air qu'ils respirent : ingrat par nature, l'utilisateur, est peu conscient des efforts qui ont été nécessaires pour son confort.

À partir d'un certain stade, cela ne vaut plus la peine de lancer de nouveaux grands projets : leur apport serait inférieur au coût des perturbations qu'ils provoqueraient⁸. Il faut donner la priorité à la bonne utilisation du système d'information existant. Alors les pionniers qui l'avaient mis en place s'ennuient.

3. Dans l'industrie du début du XIX^e siècle la liste des « moteurs » était la suivante : moulins (à vent, à eau, à manège), machines à vapeur, chevaux et mulets, bœufs (Statistique générale de la France, *Industrie 1847*).

4. Au sens des langages à objets : le « composant » est un ensemble de classes qui, dans le système d'information, représente un être du monde réel ou fournit un service.

5. Le « bus » est un « middleware » qui traite des questions importantes tant au plan fonctionnel qu'au plan technique : compatibilité des codages, synchronisme, concurrence, persistance, performance, accès à des services commun (sécurité, supervision, métrologie etc.).

6. L'interopérabilité avec les partenaires suppose non seulement que les systèmes d'information soient de bonne qualité, mais aussi que l'entreprise ait pratiqué l'« ingénierie d'affaires » convenable pour partager les responsabilités, coûts et recettes entre elle et ses partenaires.

7. Le SI est cohérent, pour le client externe de l'entreprise comme pour l'utilisateur interne, quel que soit le média utilisé (courrier, téléphone, Internet, « présentiel », carte à mémoire etc.). C'est une des conditions du commerce électronique.

8. De même, dans une ville, on doit trouver l'équilibre entre les chantiers et la vie courante : les chantiers visent une amélioration future mais perturbent le fonctionnement aujourd'hui.

Peut-être faut-il une relève : les hommes qui savent faire fonctionner un système existant n'ont pas le même caractère que ceux qui ont lutté pour « changer le monde⁹ ».

* *

Que reste-t-il à faire quand le système d'information est mûr ? Il faut d'abord le faire vivre, l'utiliser ; cela suppose de former les nouvelles recrues, d'animer la diffusion des « bons usages », d'administrer les droits d'accès, d'administrer aussi les processus pour conforter la qualité des produits. Cela représente une quantité de travail non négligeable.

Puis il faut faire en sorte que le système d'information se maintienne au niveau de l'état de l'art. Or celui-ci évolue, tant du point de vue fonctionnel que du point de vue technique. La veille technologique, réalisée par la direction informatique, doit permettre de s'assurer que l'entreprise utilise convenablement les possibilités offertes par les TIC¹⁰ ; la « veille SI », réalisée par les maîtrises d'ouvrage, permet de voir la façon dont les entreprises comparables utilisent leur système d'information.

Il faut aussi surveiller deux frontières et éventuellement les réviser :

- la frontière qui sépare la part du système d'information confiée à un ERP et celle qui est programmée « en spécifique » : elle change en raison de l'évolution des ERP ;

- la frontière entre la part du système informatique exploitée en interne et celle qui est externalisée : elle change en raison de l'évolution de l'offre d'hébergement.

Ensuite il faut adapter le système d'information aux évolutions de l'entreprise et du marché : produits nouveaux, nouvelles clientèles, nouveaux procédés de production et de commercialisation, nouveaux partenaires.

Ainsi même si le flux des projets nouveaux est plus faible que lors de la mise à niveau il ne s'est pas tari.

On doit enfin se soucier de minimiser le coût du système d'information, souci toujours présent mais qui avait pu passer au second plan pendant sa mise en place. Il est temps de l'industrialiser. Les projets étant désormais moins nombreux, la dépense porte essentiellement sur l'exploitation et la maintenance.

9. Alors que j'étais responsable de la maîtrise d'ouvrage dans une grande entreprise, j'ai eu avec son DSI un échange qui m'a paru révélateur :

DSI - « Je ne comprends pas ce que tu cherches à faire ; quel est le but que tu poursuis ? »

MV - « Je cherche à faire en sorte que le système d'information équipe convenablement les métiers, que l'entreprise soit efficace. »

DSI - « Holà, tu veux donc changer le monde ! »

MV - « Oui, modestement bien sûr ; quel autre but peut-on poursuivre dans la vie ? »

DSI - « Mon but à moi, c'est de faire du business. »

10. Ces possibilités vont évoluer rapidement au moins tant que la loi de Moore jouera, c'est-à-dire au moins jusqu'en 2015 ou 2020.

14.5 Études de cas

14.5.1 Le système d'information d'IBM

Dans la série des « failure stories », voici l'exemple d'IBM. La description sans complaisance qu'en donne Gerstner [65] illustre l'adage selon lequel les cordonniers sont les plus mal chaussés mais les mesures qu'il a prises ont semble-t-il porté des fruits.

Dirigeant, si le système d'information de ton entreprise est lamentable, ne désespère pas ! Tu es loin d'être seul dans ce cas : des entreprises beaucoup plus grosses et *a priori* plus expertes que la tienne en sont au même point ou pire encore. Il n'est jamais trop tard pour redresser la situation. Les mesures nécessaires ne demandent *que* de la volonté, du courage, du discernement - et aussi de l'argent, certes, mais il sera vite récupéré.

* *

Arrivant à IBM en 1993, Gerstner pensait trouver le meilleur système d'information du monde. Il découvrit qu'il n'en était rien. IBM dépensait quatre milliards de dollars par an pour son système d'information sans pour autant disposer des informations nécessaires pour faire tourner l'entreprise.

Les applications de comptabilité, gestion des stocks, production et livraison, programmées au cours de l'histoire, avaient bourgeonné pour répondre aux besoins de 24 directions indépendantes. Elles étaient incapables de communiquer entre elles. IBM avait 155 centres informatiques et 31 réseaux informatiques éparpillés dans le monde, dont plusieurs inactifs ou inefficaces.

128 personnes portaient le titre de CIO (« Chief Information Officer », l'analogue américain de notre DSI). Chacune d'entre elles gérait sa propre plate-forme technique et finançait le développement de ses propres applications. Les normes, les spécifications étaient choisies de façon indépendante et elles étaient donc souvent incompatibles, comme dans les chemins de fer du XIX^e siècle. Les processus étaient compliqués, redondants et coûteux. IBM avait par exemple 266 applications de tenue des comptes, ce qui rendait impossible la consolidation des données financières. La gestion des ressources humaines était si rigide que pour muter une personne d'une division à l'autre il fallait formellement la licencier de la première division, puis la faire embaucher par la seconde.

Pour corriger ces travers, Gerstner a lancé une des plus grandes opérations de « business process reengineering » de l'histoire, avec plus de 60 projets couvrant tous les domaines de l'entreprise.

IBM a maintenant un seul CIO ; le nombre des centres informatiques a été réduit à 16, le réseau a été unifié. Les dépenses en système d'information ont été réduites de moitié et plusieurs milliards de dollars annuels ont été économisés en améliorant les processus.

14.5.2 Le système d'information du FBI

On imagine que les agents du FBI, qui ambitionnent de faire trembler les malfaiteurs, disposent des meilleurs outils que la technologie puisse fournir et utilisent des méthodes conformes aux meilleures pratiques professionnelles. C'est ainsi que les films de fiction nous les montrent. En réalité il n'en est rien, comme le prouve un rapport du ministère de la Justice américain¹¹.

Le titre de ce rapport révèle un biais : l'audit porte sur les investissements, donc sur les projets et non sur le système d'information lui-même ni sur la façon dont il est utilisé. C'est un travers souvent observé : on accorde plus d'importance aux projets qu'au système existant. Malgré ses limites, cette approche fournit des indications sur l'état du système d'information.

* *

Le FBI emploie 27 000 personnes ; son budget a été en 2002 de 4 371 millions de dollars, dont 714 millions pour le système d'information. Cependant sa plate-forme technique, obsolète, ne répond pas aux besoins des utilisateurs : en septembre 2000 plus de 13 000 PC, ayant de quatre à huit ans d'âge, étaient incapables de faire tourner les logiciels de base ; certains des réseaux avaient plus de douze ans.

Les applications ne sont ni conviviales, ni capables de communiquer *via* la Toile : par exemple la messagerie fonctionne mal. Les procédures de sécurité (contrôles, authentification, sauvegardes etc.) sont déficientes. L'utilisation des bases de données est difficile et pénible pour les agents. Le FBI n'a pas de référentiel.

N'ayant pas urbanisé son système d'information, le FBI le connaît mal. Il ne s'est doté ni d'un comité stratégique des systèmes d'information, ni d'une méthode de gestion du portefeuille applicatif, ni d'une procédure de sélection : les diverses divisions entassent les expressions de besoin sans avoir connaissance des priorités et besoins globaux. La direction financière et la direction informatique croient chacune que l'autre est chargée de la sélection des projets.

Le FBI consacre ainsi des centaines de millions de dollars à des projets informatiques sans pouvoir s'assurer qu'ils répondent à leurs objectifs. Les projets ne sont pas vraiment suivis. Ils souffrent d'importants dépassements en délai et en coût en raison des déficiences de leur gestion. Les tout premiers efforts pour améliorer l'organisation ont été engagés... au moment du lancement de l'audit.

Le processus d'investissement est jugé « incohérent, mal structuré et aléatoire ». L'attention se concentre sur les dépenses d'investissement et néglige les dépenses d'exploitation et de maintenance. Les responsabilités sont mal définies, les compétences nécessaires ne sont pas mobilisées, les décisions sont prises à la va-vite.

11. « Federal Bureau of Investigation's Management of Information Technology Investments », *United States Department of Justice, Office of the Inspector General, Audit Division*, décembre 2002, www.usdoj.gov/oig/audit/0309/0309.pdf

L'audit a examiné le projet Trilogy qui vise à améliorer la plate-forme informatique, les réseaux et les postes de travail sans ambitionner toutefois de rattraper le niveau de l'état de l'art. Ce projet, dont le coût prévisionnel était de 380 millions de dollars, a été lancé en 2001 en vue d'une mise en exploitation en 2004. Il a été accéléré après l'attentat du 11 septembre 2001 et son budget a été porté à 458 millions de dollars, mais la livraison prévue pour juillet 2002 a été reportée à mars 2003 et il faudra encore accroître le budget pour réaliser le poste de travail, partie la plus importante du projet.

Les échéances ont été souvent révisées. Les études préliminaires avaient été trop superficielles : d'importantes options d'architecture comme le client léger ou l'utilisation de la Toile ont dû être abandonnées lorsque leurs implications techniques se sont révélées trop lourdes.

Pour définir le poste de travail il faudra faire l'inventaire du système d'information existant, ce qui oblige à un lourd « reverse engineering ». Le déploiement des nouveaux PC, la mise en place des nouveaux réseaux se font dans le désordre. Enfin le FBI n'a pas mis à jour son plan stratégique depuis 1998, et le plan existant ne comporte rien sur le système d'information.

* *

Ces défauts sont connus depuis longtemps, mais il a fallu le 11 septembre 2001 pour que les responsables se décident à lancer un audit et, parallèlement, à faire un premier progrès dans les méthodes. Le rapport d'audit est d'une grande clarté, et il est publié sur la Toile : si le système d'information du FBI est défectueux, la publication de ce rapport témoigne d'une transparence qui annonce un redressement.

Cinquième partie

L'informatique dans le
fonctionnement

Chapitre 15

Économie de l'informatique

Dans leur majorité les entreprises ne font pas encore beaucoup d'efforts pour connaître le coût de leur informatique. Cela surprend si l'on se rappelle que l'étude détaillée des coûts de production, puis l'effort pour les réduire, ont été l'un des moteurs les plus puissants des progrès de l'industrie. Tout se passe comme si la dépense informatique était une fatalité à laquelle on ne peut opposer que deux attitudes également aveugles : soit un laisser-aller que l'on croit nécessaire au suivi de l'état de l'art, soit une politique de « réduction des coûts » sans discernement.

Cet aveuglement n'est plus tenable dans telle grande entreprise de ma connaissance dont l'informatique représente 15 % des coûts, pour un montant annuel de deux milliards et demi d'euros, et où les salariés passent plus de la moitié de leur temps de travail devant l'ordinateur en réseau. Lorsqu'on en est là il importe non seulement d'évaluer le coût de l'informatique, mais de disposer d'un modèle permettant de maîtriser la dynamique des projets, la durée de vie des produits, l'accumulation des travaux de maintenance ainsi que les coûts cachés que comporte la mise du poste de travail à la disposition des utilisateurs.

L'élaboration de ce modèle supposera souvent un retraitement des données comptables pour redresser des biais systématiques : à la comptabilité fondée sur la trésorerie ou la facturation devra être préférée la comptabilité au fait générateur, même si celle-ci nécessite des estimations ; la frontière entre dépense d'exploitation et immobilisation devra être redéfinie en fonction de critères économiques et non selon la réglementation fiscale ; l'évaluation de l'annuité équivalente à une dépense d'immobilisation devra tenir compte de la durée de vie anticipée de l'actif ; le coût associé à une décision en informatique devra couvrir l'ensemble des dépenses qu'elle occasionne, même lorsqu'elles sont supportées par d'autres entités de l'organisation (coûts de la maîtrise d'ouvrage, du déploiement, de la formation) etc.

La clarté sur les coûts et sur leur dynamique responsabilise les métiers de l'entreprise et en premier la direction informatique elle-même. Celle-ci, responsable de la gestion de la plate-forme technique, doit régler divers « curseurs » selon son anticipation des coûts, de la pérennité des fournis-

seurs et des évolutions de l'état de l'art ; elle doit distinguer les compétences à posséder en interne, et celles que l'on achètera à des SSII ; puis les produits qu'il faut réaliser de façon spécifique, et ceux que l'on peut obtenir en paramétrant un progiciel ; enfin les travaux à externaliser, et ceux qu'il vaut mieux faire soi-même. Elle doit aussi maîtriser la pente qui, partant du classement des projets selon leur coût, risque de privilégier la solution la plus coûteuse parce qu'elle est la seule visible pour le directeur.

La connaissance des coûts modifie les priorités du stratège. Dans l'entreprise qui ignore ses coûts l'attention du DG se focalise sur les projets, épisodes glorieux de l'entreprise. Mais les projets ne représenteront parfois que 10 % du coût de l'informatique alors que le réseau de postes de travail en représente 50 %, la maintenance et l'exploitation du « stock » applicatif 40 %. Mieux vaut alors, avant de lancer de nouveaux projets, animer la bonne utilisation des outils existants et veiller à la satisfaction des utilisateurs.

Au début des années 2010, les salariés français passeront plus de la moitié de leur temps de travail devant l'ordinateur en réseau (voir page 201). L'enjeu principal des entreprises réside donc déjà, même si elles ne le savent pas toutes, dans la définition du pivot qui articulera la capacité de synthèse et de décision de l'être humain organisé à la souple puissance de l'automate programmable. La réflexion sur le coût de l'informatique est alors un préalable à la maîtrise du coût des processus de production : les décisions relatives à l'automate ne se séparent pas de celles, qui lui sont duales, concernant la gestion de la ressource humaine et de ses compétences.

Ainsi la « physique » de l'entreprise, c'est-à-dire la définition pratique de son organisation interne comme de son articulation avec le monde extérieur, associée à la définition sémantique (et donc symbolique) du système d'information, revient au premier rang. Cela rappelle ces années 1880 lors desquelles les usines furent transformées par l'apparition du moteur électrique, et le travail de bureau par le changement d'échelle qui s'amorça dans le « loop » de Chicago.

15.1 Enjeu macroéconomique

Les économistes ont du mal à évaluer l'enjeu que représente le système d'information des entreprises.

Ils perçoivent sans doute le retard de l'Europe par rapport aux États-Unis dans les TIC ; mais lorsqu'ils cherchent à définir la politique qui permettrait de le combler, leur intuition les oriente non vers les systèmes d'information mais vers les techniques fondamentales de la microélectronique et du logiciel (voir page 21) - voire vers celles, futuristes, de la nanotechnologie et de la bioélectronique - ou encore vers la production de machines fondées sur ces techniques (ordinateurs, routeurs etc.).

Or dans ces deux domaines-là l'Europe aura beaucoup de mal à rattraper son retard. Il faudrait mettre en place, de façon volontariste, un réseau de coopération entre universités, centres de recherche et entreprises, puis encore financer ce réseau pendant des années avant qu'il puisse être compétitif. Un tel effort est certes souhaitable, mais est-il possible alors que l'on cherche à

compresser la dépense publique? Les échecs passés (Plan Calcul etc. : voir Brulé [28]) incitent au scepticisme envers les politiques publiques dans ce domaine.

Il existe cependant un domaine où l'économie européenne, l'économie française pourraient progresser sans qu'il fût besoin de lancer un programme macroéconomique de dépense publique et de politique industrielle représentant quelques pourcents du PIB : c'est celui des systèmes d'information, de l'*utilisation* des TIC.

C'est là en effet que réside l'enjeu économique principal. Donner des microprocesseurs rapides, des mémoires volumineuses, des réseaux à haut débit, des logiciels performants à une entreprise qui ne sait pas les utiliser, c'est comme donner une moto puissante à une personne qui ne sait pas conduire : elle aura tôt fait d'avoir un accident.

L'efficacité des TIC ne se dégage que si l'on sait utiliser convenablement les outils qu'elles fournissent. Or ce savoir-faire est difficile à acquérir. Les entreprises qui ont cru qu'il suffisait d'acheter des machines, réseaux et logiciels ont été déçues. Certains en ont déduit, trop vite, que les TIC étaient inefficaces¹ : elles ne le sont que quand on les met entre des mains maladroites.

L'erreur la plus fréquente consiste à faire comme si l'utilisation des TIC était une affaire essentiellement technique. L'informaticien est alors considéré comme un pur technicien : on lui demande de faire fonctionner une machine, sans pannes et sans que l'entreprise n'ait à s'en préoccuper. Cette erreur, notons-le, est commise par des personnes qui savent bien par ailleurs que l'informatique transforme l'entreprise, qu'elle modifie la pratique des métiers comme le domaine du possible : mais il y a loin entre cette intuition vague et la maîtrise pratique, opératoire, qui seule permettrait de savoir s'y prendre. Un adolescent peut se passionner pour les avions, s'en faire une idée qui n'est pas fautive : il lui reste beaucoup à apprendre pour en piloter un. Certaines personnes, ayant un goût sûr en peinture, apprécient à leur juste valeur les tableaux les mieux réussis ; mais quand elles entreprennent de peindre elles ne produisent que des croûtes.

La maîtrise du système d'information suppose :

- que l'entreprise ait explicité son langage, ses concepts, nomenclatures, classifications, identifiants, définitions, segmentations, organisations, de façon à pouvoir les faire évoluer ;
- qu'elle sache distinguer les tâches qu'elle confiera à l'automate de celles qui doivent être exécutées par des êtres humains ;
- qu'elle ait impliqué l'automate dans ses divers processus : approvisionnement, production, commercialisation, distribution, documentation, gestion ;
- que ses processus soient élucidés par des indicateurs et tableaux de bord ;
- que l'entreprise elle-même soit élucidée, le système d'information four-

1. Cette erreur de perspective est à l'origine du « paradoxe de Solow » (voir page 181).

nissant aux salariés, dirigeants, partenaires etc. une vue claire de ce qu'elle est et de ce qu'ils ont à faire.

« Éclaircir », « clarifier », tout le monde peut comprendre ces objectifs - et lorsqu'ils sont atteints tout paraît simple. Mais pour les atteindre il faut une bonne maîtrise des procédés de pensée, de la sémantique, de l'articulation entre le langage et l'action : l'entrepreneur efficace est ici un *praticien de la philosophie de l'action*.

Les économistes parlent de « rationalisation », « optimisation », « maximisation du profit » etc., mais beaucoup d'entre eux ignorent les conditions pratiques auxquelles est soumise la poursuite de ces objectifs. Les *managers* s'appuient sur la tautologie du *business is business* et confortent leur légitimité en répétant des formules de consultant : « *business process reengineering* », « *bottom line* », « créer de la valeur pour l'actionnaire » etc. Il y a loin entre la trivialité de ces slogans et la finesse du discernement nécessaire !

Il est possible de conduire une politique économique en matière de système d'information : il s'agit de définir et de partager de « bonnes pratiques ». Cela passe par l'exemple et l'animation. Une telle action suppose de l'intelligence, mais ne réclame pas un budget d'ampleur macroéconomique.

L'exemple peut être donné par les grandes entreprises dont certaines ont déjà bien avancé dans la maîtrise du système d'information, et que rassemblent le CIGREF (Club Informatique des Grandes Entreprises Françaises)² et le Club des maîtres d'ouvrage des systèmes d'information³. Il faudrait les inciter à diffuser des analyses, des études de cas, des monographies.

L'exemple peut être donné aussi par les administrations : une préfecture est le nœud d'un grand nombre de processus administratifs. L'administration fiscale, la sécurité sociale peuvent éclaircir leurs processus, articuler de façon raisonnable l'automate et l'être humain (c'est l'objet du projet Copernic du ministère de l'économie et des finances dont la toute première démarche, fait significatif, a consisté en une mise en ordre des identifiants).

Les entreprises qui ont mis en place une maîtrise d'ouvrage professionnelle et qualifiée auront gagné en efficacité en même temps qu'elles donnent le bon exemple : c'est ainsi, par contagion et imitation, que se déclenchent les effets de boule de neige qui font mûrir les conséquences d'une innovation.

Un tel enjeu justifierait la mobilisation des politiques comme des économistes qui les conseillent. Se mobiliseront-ils ? Considérons le cas de la CNAM-TS⁴. L'enjeu de son système d'information est d'ampleur macroéconomique : il s'agit de la maîtrise du déficit public, du respect des accords de Maastricht, et bien sûr aussi de la qualité du système de santé. Le gouvernement, justement alarmé par le « trou de la sécu », fera-t-il figurer dans la liste de ses mesures prioritaires l'amélioration du système d'information de

2. www.cigref.fr

3. www.clubmoa.asso.fr

4. Caisse nationale d'assurance maladie des travailleurs salariés.

la CNAM-TS? Cette opération, plusieurs fois annoncée mais jamais réalisée, sera-t-elle enfin lancée?

J'ai entendu des économistes dire « si l'on améliore le système d'information de la CNAM-TS, cela fera croître encore davantage les dépenses de santé et le trou de la sécu ». C'est une nouvelle formulation du paradoxe de Solow : « Plus le système d'information est efficace, moins l'entreprise est efficace » !

Nier contre l'évidence les apports du système d'information permet de se complaire dans la démarche habituelle de la politique économique : régler et consacrer des budgets importants à de grands projets. L'échec ne se manifestant qu'après un délai cette méthode permet d'empocher un effet d'annonce. Elle permet aussi, en refusant de voir l'évolution de nos entreprises, d'éviter l'effort nécessaire pour la comprendre.

Les habitudes des économistes et des politiques pèsent dans un sens ; les besoins et l'expérience des entreprises pèsent dans l'autre. Certaines personnes, placées à la charnière des deux mondes, s'efforcent de bousculer les habitudes en faisant valoir l'urgence des besoins. Rien n'est perdu, comme rien n'est gagné, mais plus nous serons nombreux à savoir que le système d'information est un enjeu macroéconomique primordial mieux cela vaudra.

15.2 Un facteur de production

Le système d'information est pour l'entreprise un facteur de production et lorsqu'on parle d'un facteur de production, l'économiste doit considérer son coût, sa contribution à la fonction de production et la façon dont il est géré. Mais la plupart des entreprises considèrent non le coût de leur système d'information mais celui de leur seule informatique : les dépenses de la maîtrise d'ouvrage (expression de besoins, spécifications fonctionnelles, suivi du projet, recette fonctionnelle, formation des utilisateurs, déploiement sur le terrain, suivi de la mise en œuvre) sont souvent négligées alors qu'elles représentent de l'ordre de 20 % du coût du système d'information.

À l'intérieur du budget informatique lui-même, l'attention apportée aux divers postes est variable. Elle se focalise volontiers sur les projets nouveaux - sur lesquels il est possible de faire des économies - et du coup certaines proportions échappent à l'intuition des décideurs.

Indiquons donc des ordres de grandeur : le budget informatique annuel d'une entreprise de services de 50 000 salariés est de l'ordre de 250 millions d'euros, soit 5 000 € par personne et par an. Les deux tiers de cette somme sont consacrés aux postes de travail en réseau (PC, serveurs locaux, routeurs, réseaux LAN et WAN, formation et assistance aux utilisateurs). Il reste environ 1 700 € par personne et par an pour les serveurs centraux, la maintenance du parc applicatif et les projets nouveaux.

Supposons que la part des projets nouveaux soit le tiers de ce total : elle ne représentera qu'un dixième du coût total de l'informatique, soit 25 millions d'euros dans le cas considéré. C'est une somme non négligeable mais en concentrant l'attention et la discussion sur elle on risque de négli-

ger d'autres postes plus importants : dépenses de la maîtrise d'ouvrage (50 millions d'euros), réseau des postes de travail (170 millions d'euros), coût de la maintenance (que l'on considère souvent comme une fatalité).

On voudra bien pardonner les approximations grossières que comporte le calcul ci-dessus. Dans chaque entreprise, une évaluation précise est possible, mais difficile car les conventions comptables introduisent de la confusion. Ainsi, on appelle « TMA » (tierce maintenance applicative) la fonction des fournisseurs qui assurent la maintenance du parc d'applications ; mais leurs contrats comportent, outre la maintenance au sens strict (« MCO », maintien en condition opérationnelle), des développements nouveaux (« maintenance évolutive ») jugés trop petits pour faire l'objet de projets en bonne et due forme. Et si la DSI dispose d'un budget de TMA élevé, elle sera tentée de faire passer discrètement, sous le vocable « maintenance évolutive », des modifications relativement importantes qui auraient pu faire l'objet d'un projet.

Enfin la diversité des sources de financement est une autre cause de confusion. Les projets prenant souvent du retard, certains budgets sont reportés d'une année à l'autre. L'affectation des dépenses à une année sont faites selon des méthodes diverses qui parfois se mêlent : flux de trésorerie (date de paiement des factures) ; enregistrement comptable (date de réception des factures) ; service fait (date d'approbation des factures) ; fait générateur (estimation au prorata de l'avancement des travaux).

L'évolution technique rend vite obsolètes les « règles de pouce » usuelles et les décideurs doivent réviser périodiquement leurs habitudes : le coût des matériels diminue rapidement (à performance égale) et la solution raisonnable une année ne le sera plus quelques années après ; l'offre de passerelles et autres *middlewares* évolue elle aussi, ce qui rend obsolètes les choix d'architecture les mieux ourdis. Il faut sans cesse arbitrer entre le souci de préserver la stabilité de l'édifice et celui de le maintenir au meilleur niveau de performance.

* *

Il est difficile d'évaluer la rentabilité d'un investissement dans le système d'information. L'anticipation de son coût est incertaine et l'estimation du résultat que l'on peut attendre de l'investissement l'est plus encore : s'il s'agit d'un gain de productivité, sera-t-il réel ou devra-t-on conserver des personnels peu productifs ? s'il s'agit d'un gain de part de marché, pourra-t-on le réaliser ? Nous avons tous entendu un directeur général dire « si le système d'information avait permis les gains de parts de marché que l'on m'a promis, nous aurions depuis longtemps dépassé les 100 % »...

Cependant sans système d'information l'entreprise s'arrêterait : aucun transporteur aérien ne peut survivre sans système de réservation, aucune banque ne peut vivre sans gestion informatique des comptes, aucun opérateur télécoms ne peut se passer de la commutation électronique etc.

Lorsqu'un facteur de production est vital pour une entreprise, sa rentabilité est en pratique infinie - c'est d'ailleurs ce qui rend l'estimation de celle-ci pratiquement impossible⁵.

* *

Le système d'information, en tant que facteur de production, relève de la catégorie du capital : c'est un *stock* qui s'accumule d'abord et qui s'utilise ensuite dans la durée. Comme tous les autres types de capital, il s'use : chaque année, une partie des applications devient obsolète. On peut donc appliquer à l'informatisation les méthodes de la micro-économie, la notion clé étant « l'intensité capitaliste », volume du capital par tête. On peut évaluer le *degré d'informatisation* d'une entreprise en considérant le « volume du système d'information » par utilisateur (qu'il soit mesuré en nombre de points de fonction, en nombre de lignes de code source ou autrement, peu importe ici).

Le calcul (voir page 537) montre alors qu'il existe un degré d'informatisation optimal : si l'entreprise ne l'atteint pas, elle est sous-informatisée ; si elle le dépasse, elle est sur-informatisée. Il serait cependant en pratique difficile d'utiliser la formule que fournit le calcul : le bon niveau d'informatisation doit être atteint par tâtonnement et en usant du bon sens. La solution efficace ne sera en tout cas jamais fournie par une politique extrême (refus de l'informatisation ou informatisation forcenée).

Ce que nous venons de dire suppose que diverses questions d'efficacité aient été résolues au préalable, comme c'est l'usage en microéconomie. Elles ne sont pas faciles pour autant. Chacune des couches du système d'information pose des problèmes de qualité et d'efficacité spécifiques, leur fonctionnement conjoint en pose encore d'autres. Ce n'est qu'après avoir résolu ces problèmes, ainsi que ceux que posent la construction des programmes et le déroulement des projets, qu'il convient de s'interroger sur le degré d'informatisation optimal.

* *

La nature et l'ampleur de l'évolution que le système d'information apporte aux entreprises sont difficiles à se représenter. Les historiens futurs diront que nous avons vécu l'âge de l'informatique mais si nous en avons quelque peu conscience il nous est difficile d'en tirer des conclusions : sans doute en était-il de même pour ceux qui ont vécu sans le savoir l'âge de la machine à vapeur (sans parler de ceux qui ont vécu l'âge de la pierre taillée, les premiers à tirer parti des propriétés électroniques du silicium).

La part du tertiaire dans l'emploi est largement majoritaire dans toutes les économies riches : en France, ce secteur représente plus de 75 % des emplois et sa part continue à croître sans fléchir (voir le tableau 5.1 page

5. Il m'est arrivé de dire, lors de la présentation d'un projet qui dégageait dès la première année un flux de trésorerie positif, « dans ce cas là il est impossible d'évaluer la rentabilité » : en effet elle est infinie. Parler en mathématicien était une erreur : le directeur financier, énarque distingué, en a conclu que le projet n'était pas rentable et malgré mes protestations il en a convaincu le comité de direction.

201). Or le poste de travail classique d'un employé du tertiaire, c'est un bureau, un fauteuil, une armoire et un PC en réseau⁶. Le coût annuel total du PC en réseau représente l'équivalent du coût des m² de bureau occupés par l'employé (s'il faut 15 m² par personne en moyenne et si le loyer est de 250 € par m² et par an.)

Le réseau confère l'ubiquité à l'automate mais cette ubiquité est en train de changer de nature. Elle signifie aujourd'hui que l'on peut, de tous les postes de travail de l'entreprise, accéder au même système à condition d'avoir les habilitations requises. Mais le téléphone mobile, l'ordinateur et le *palmtop* se préparent à fusionner dans un boîtier que l'on portera à la ceinture, les lunettes ou d'autres supports servant d'écran. Alors l'ubiquité logique sera absolue : ce ne sera plus le bureau, mais le *corps* lui-même qui sera équipé⁷, chacun accédant à une ressource personnelle de puissance et de mémoire fournie par des serveurs dont il ne se soucie pas de connaître la localisation.

Les exigences auxquelles le système d'information doit répondre sont plus strictes que dans la phase antérieure, celle des grandes applications de gestion, qui semble rétrospectivement avoir été quelque peu rustique. Pour outiller les activités et processus de production le système d'information doit reposer sur la modélisation des composants concernés ; la gestion des identifiants et nomenclatures doit être rigoureuse, ainsi que l'utilisation des tables de référence. Un même système d'information doit par ailleurs fournir des vues diverses sur l'entreprise : la vue opérationnelle des agents qui traitent les dossiers un par un ; la vue qui sert au pilotage par le management opérationnel, lui fournissant les indicateurs pour contrôler la qualité des processus et la bonne utilisation des ressources ; enfin, les données statistiques qui alimentent la réflexion stratégique sur la concurrence, le positionnement, les évolutions du marché, la segmentation de la clientèle.

Ce système d'information partout présent doit encore être *sobre* : il ne conviendrait pas de faire avec l'informatique ce qui peut se faire efficacement avec un calepin et un crayon ; ni de recourir au système d'information central pour un calcul que l'on peut faire localement sur un tableur ; ni de fournir des fonctionnalités dont 80 % ne seront jamais utilisées.

Nos entreprises sont enfin plongées dans un monde marqué par la *nouvelle économie*, elle-même caractérisée par les rendements croissants, la concurrence monopoliste, la différenciation des produits (Volle [213]). Elles doivent s'allier avec des partenaires, articuler l'Internet à leur activité commerciale, entretenir une relation multimédia avec leurs clients, partenaires et fournisseurs, fournir à leurs clients une aide pour trouver le produit qui répond à leurs besoins, leur fournir aussi des services d'avant et d'après-vente. Tout cela n'est possible que si le système d'information est de bonne qualité.

6. Les vendeurs dans les grands magasins ne sont cependant généralement pas équipés de la sorte.

7. Tout comme aujourd'hui le téléphone mobile équipe le *corps* et non plus le bureau ou l'appartement.

Si le système d'information est désordonné il sera difficile de conclure des partenariats stables, car on ne pourra pas procurer au partenaire la transparence qui garantirait qu'il ne se fait pas gruger. L'expérience montre que des partenariats conclus sans que l'on ne se soit soucié de l'interopérabilité des systèmes d'information ne durent guère.

De même, il sera difficile pour une entreprise de se « mettre à l'e-business », de « faire du B2B et du B2C » si elle n'a pas un bon système d'information. J'ai connu une entreprise industrielle qui s'était développée en achetant dans divers pays d'autres entreprises du même secteur. La seule contrainte qu'elle leur avait imposée en matière de système d'information était de fournir le reporting financier relatif au mois m dès le jour 2 du mois $m + 1$. Les catalogues, les processus de traitement des commandes avaient été laissés tels quels : pour pouvoir présenter son offre sur l'Internet elle a dû d'abord mettre tout cela en ordre, tâche immense qui a imposé au stratège un délai fort contrariant.

* *

Voici sans souci d'exhaustivité quelques exemples des pièges que rencontrent les entreprises :

1) Une entreprise du tertiaire travaille depuis trois ans pour « dématérialiser » son offre de services, en l'occurrence faire passer du papier à l'Internet la relation avec ses clients. Les consultants ont été consultés ; les groupes de travail ont travaillé ; les cadres ont fourni leur expertise ; les clients ont exprimé leurs demandes ; les plans sont prêts.

On sait où aller et cependant rien ne bouge : les directeurs se faisant la guerre, chacun change d'avis lorsqu'il pourrait arriver à un accord avec les autres. Un des groupes de travail a dit qu'il fallait une « conduite du changement », ce qui signifie que l'on va s'efforcer de faire accepter par les personnels les changements de l'organisation et des procédures de travail. Cette entreprise n'aurait-elle pas plutôt besoin d'un « changement de la conduite » ?

2) Un consultant a conseillé au chef d'un projet de système d'aide à la décision d'utiliser les données vérifiées par le service statistique plutôt que de faire faire en double les vérifications et redressements. « Non, répondit celui-ci, car alors je serais dépendant d'eux ».

L'image qui s'impose est celle d'une ville où les immeubles ne s'appuieraient pas l'un sur l'autre et auraient chacun ses quatre murs porteurs, et en outre un puits pour s'approvisionner en eau, un groupe électrogène etc.

3) Plusieurs dirigeants de grandes entreprises françaises, longtemps rétifs devant l'Internet (« cette mode, ce gadget futile etc. »), ont fait vers la fin de 1999⁸ un pèlerinage aux États-Unis. Ils en sont revenus illuminés : il leur fallait un *e-business* tout de suite, dans six mois au plus.

Une de ces entreprises avait dans le monde 800 commerciaux qui connaissaient bien leurs acheteurs et savaient comment les fidéliser. « Qu'allez-vous faire de vos commerciaux quand vous vendrez sur l'Internet ? » demanda

8. C'est-à-dire quelques mois avant l'éclatement de la « bulle »...

un consultant. « Nous allons nous débarrasser d'eux », répondirent ingénument les dirigeants. Ils n'avaient pas pensé que les commerciaux pourraient dans l'intervalle dénigrer l'Internet auprès des clients, ni que l'Internet ne pourrait pas remplacer instantanément, comme par miracle, une expertise commerciale qui était l'un des acquis les plus précieux de leur entreprise.

4) Dans une grande entreprise de service, les dirigeants passent une partie de leurs réunions à confronter des indicateurs incohérents et à s'en expliquer les différences : « Nous n'avons pas encore pris en compte le changement de limites des régions », « notre nomenclature de produits n'est pas à jour », « on nous impute des dépenses que nous ne maîtrisons pas » etc.

Leur énergie s'use devant des tableaux de bord mal conçus ; mais ils auraient du mal à définir le tableau de bord dont ils ont besoin, et ignorent tout des subtilités intellectuelles que requiert la maîtrise des données.

5) Une entreprise met en place un *datawarehouse* et en même temps ses divers processus opérationnels se dotent d'infocentres. Elle court le risque de faire deux fois la même chose, et aussi le risque de ne pas répondre aux besoins : chacun des projets croit que l'autre s'en charge et la balle passe entre les deux joueurs comme dans une partie de tennis en double.

6) Nous travaillons dans l'économie de l'immatériel qui est aussi une économie de la compétence, du savoir ; nous pratiquons le *knowledge management* etc. : cela n'a pas empêché nos entreprises de faire partir les cadres de plus de cinquante ans, car c'était la mode. Elles ont alors perdu des compétences dont la reconstitution leur demandera plusieurs années.

7) Pour que puissent coopérer les spécialités pointues dont l'entreprise a besoin dans l'économie de la compétence, il faut que les spécialistes soient capables d'écoute et de dialogue, qu'ils sachent respecter les personnes relevant d'autres spécialités. Il faut aussi un dialogue entre expert et décideur : ces deux fonctions ne peuvent pas être tenues par la même personne. On est ici aux antipodes du corporatisme défensif comme de l'autoritarisme.

Ces problèmes là existent. On peut préférer ne pas les voir et idéaliser l'entreprise en prétendant qu'elle est strictement rationnelle. Ce n'est pas ainsi que l'on guérira les maladies dont elle souffre. Respecter l'entreprise, c'est au contraire la considérer comme un être vivant et fragile dont la santé nécessite une approche médicale (voir page 543).

15.2.1 Évaluer une entreprise

À la Bourse, la valeur d'une entreprise est égale à sa capitalisation boursière, c'est-à-dire au produit du cours de l'action par le nombre de ses actions. Mais le cours s'établit sur l'étroit marché d'occasion où se revendent les actions : il suffit qu'un acheteur important se présente pour que le cours monte, que quelqu'un liquide un gros paquet d'actions pour que le cours diminue. La capitalisation boursière étant aussi volatile que le cours, l'évaluation qu'elle fournit est peu crédible.

Pour un comptable, la valeur de l'entreprise est égale à l'actif net des dettes, c'est-à-dire aux fonds propres. Mais pour que cette évaluation soit crédible il faudrait que la valeur de chacun des postes de l'actif (machines,

immeubles, titres, créances etc.) fût certaine. Or les règles comptables, conçues principalement pour faciliter le dialogue avec l'administration fiscale, ont un caractère conventionnel qui altère leur signification économique.

Pour un économiste, la valeur de l'entreprise est la valeur actualisée de son cash-flow⁹ futur. Cette conception de la valeur fait, contrairement aux deux précédentes, reposer l'évaluation sur une *anticipation* des données futures ; elle implique donc une part d'incertitude. Cette incertitude est prise en compte par la prime de risque que comporte le taux d'actualisation.

C'est à cette dernière méthode que j'adhère en tant qu'économiste, mais l'art de l'anticipation est délicat. Aussi il est important de disposer pour l'étayer de règles que l'on puisse appliquer aux données observables les anticipations. Ce sont de telles règles que nous allons proposer ci-dessous.

* *

Dans le futur, l'entreprise va mettre en valeur le capital fixe¹⁰ qu'elle détient aujourd'hui ; elle va investir pour le modifier ; elle va s'adapter aux modifications de l'environnement réglementaire ou concurrentiel ; elle va tirer parti des opportunités. Les critères que nous allons examiner visent à évaluer son aptitude à investir ou à évoluer de façon judicieuse.

Le but d'une entreprise est de fournir sur le marché des produits utiles à ses clients. Il faut pour cela qu'elle soit bien organisée (qualité de l'organisation), qu'elle connaisse ses clients (qualité du marketing) et qu'elle maîtrise ses moyens de production (maîtrise de la fonction de production).

Qualité de l'organisation¹¹ : la fonction première de l'entreprise est d'organiser l'action des personnes qui la composent en vue de produire des biens et services utiles aux consommateurs (directement s'il s'agit de biens de consommation, indirectement s'il s'agit de biens intermédiaires). Une entité inorganisée, qu'elle soit parasitée par des réseaux qui pompent sa substance¹² ou que chacun soit laissé libre d'y agir selon son caprice, ne mérite pas le nom d'entreprise.

L'organisation est donc l'actif le plus précieux de l'entreprise, bien qu'il ne figure pas au bilan. Sa qualité réside dans la clarté du vocabulaire et des consignes de travail partagés par le personnel ; dans l'équilibre du découpage

9. Solde des mouvements de trésorerie hors frais financiers, impôts et dividendes, et après déduction de la variation du besoin en fonds de roulement. Certains disent que la valeur de l'entreprise est la valeur actualisée des dividendes qu'elle distribuera dans le futur : du point de vue de l'économiste cette définition est erronée même si elle figure dans certains manuels.

10. Machines, réseaux, équipements, immeubles, stocks, et aussi programmes informatiques, organisation et compétences : le capital fixe se distingue du capital financier.

11. Le mot *organisation* a deux acceptions : (a) définition des missions des directions, délimitation des périmètres de la légitimité : c'est l'organisation au sens de « organisme » ; (b) définition des produits, des méthodes de travail, de l'allocation des ressources. C'est cette deuxième acception que nous considérons ici.

12. Certaines entités anciennes sont la proie de réseaux corporatifs, politiques, syndicaux ou autres qui y prélèvent richesse et pouvoir d'influence. C'est signe d'une sénilité qui, dans des conditions économiques normales, conduit rapidement à la mort, mais qui peut durer indéfiniment dans les entités bénéficiant d'un monopole de droit - notamment dans les administrations et les établissements publics.

des missions des entités, qui doit simplifier autant que possible les flux internes de produits et d'informations; dans la bonne articulation entre expertise et légitimité lors de la préparation des décisions.

Il est difficile d'évaluer directement la qualité de l'organisation, mais on peut l'évaluer indirectement en transitant par l'évaluation du système d'information, qui la reflète (cf. ci-dessous).

Qualité du marketing : pour que l'entreprise puisse fournir des produits utiles à ses clients, il faut qu'elle connaisse ceux-ci et leurs besoins. Le marketing est donc pour l'entreprise une des disciplines intellectuelles essentielles.

La qualité du marketing s'évalue en examinant la segmentation de la clientèle. Il existe des segmentations mal conçues : celles qui classent les clients selon leur opinion sur l'entreprise ne considèrent pas le client lui-même, ni ses besoins. Il arrive aussi que la segmentation ne soit pas mise à jour alors que le marché a changé.

Un marketing de qualité suppose la maîtrise de méthodes statistiques de classification, ainsi que la formation des agents opérationnels à l'utilisation des outils de diagnostic et de prescription forgés par le marketing.

Le marketing s'intéresse aussi au positionnement de l'entreprise par rapport à ses concurrents : il examine les parts de marché, en identifie les facteurs explicatifs, propose des stratégies commerciales compétitives (diversification des produits, « packaging », partenariats, tarification, fidélisation etc.).

Maîtrise de la fonction de production : la « fonction de production », c'est la relation qui existe entre les quantités d'*inputs* (facteurs de production, matières premières) et les quantités d'*outputs* (produits). Si l'on associe à chaque input son prix unitaire, on peut à partir de la fonction de production calculer la *fonction de coût* qui donne le coût de production en fonction de la quantité produite. La fonction de production évolue avec le progrès technique qui transforme méthodes et outils.

L'efficacité de la mise en œuvre de la fonction de production est le grand souci de l'entreprise : il s'agit de minimiser le coût de production et de faire évoluer la fonction de production pour utiliser des techniques efficaces, conformes à l'état de l'art.

L'entreprise connaît-elle sa fonction de production ? se tient-elle au courant de l'état de l'art, a-t-elle organisé une veille technologique et des « benchmarks » ? la mise en œuvre de sa fonction de production est-elle efficace ? les évolutions des prix des facteurs et matières premières sont-elles suivies avec attention¹³ ?

Une entreprise qui a modélisé sa fonction de production peut réaliser des simulations, des jeux stratégiques qui préparent la manœuvre à exécuter en cas de changement du prix des facteurs de production ou des matières premières, en cas de variation brusque du carnet de commande, ou encore pour diversifier l'offre sur de nouveaux segments de clientèle.

13. La maîtrise de la fonction de production suppose en particulier que les achats soient gérés avec vigilance et que les acheteurs soient soigneusement encadrés.

En principe, si les trois critères ci-dessus sont bons, le reste en découle et l'entreprise est de bonne qualité. Mais comme leur évaluation est délicate il faut examiner d'autres critères pour la confirmer et la préciser.

Qualité du système d'information : voici quelques années l'organisation résidait dans les cerveaux des personnes, se communiquait pour l'essentiel de façon orale et ne se déposait qu'en partie dans des notes dont l'interprétation était d'ailleurs délicate. Aujourd'hui le système d'information équipe toutes les procédures de l'entreprise et pour le spécifier il faut documenter l'organisation.

Alors qu'il serait difficile d'interpréter, de décoder des notes et des propos qui sont protégés à la fois par le jargon et par la prudence, le système d'information donne de l'organisation une image en quelque sorte naïve, non apprêtée : ceux qui l'ont spécifié n'ont généralement pas pensé que l'on pourrait évaluer l'organisation en considérant le système d'information. Passer par le système d'information pour évaluer l'organisation est donc une démarche efficace, même si elle est indirecte. En particulier la solidité des référentiels sera un indicateur de la cohérence de l'entreprise.

Nota Bene : Nous avons proposé page 491 une *check-list* pour évaluer le système d'information.

Qualité des produits : si le marketing est bon, si la fonction de production est maîtrisée, en principe les produits seront bons, bien positionnés sur le marché et vendus à un prix compétitif ; le réseau commercial sera efficace ainsi que le service après vente. L'évaluation de la qualité des produits permet donc de valider deux des trois critères principaux.

Qualité des compétences : la compétence de l'entreprise est le résultat conjoint de la qualité des individus qui la composent et de la qualité de l'organisation. En principe, une entreprise bien organisée attire des personnes qualifiées ; l'évaluation de la qualification des personnes permet donc de valider la qualité de l'organisation et de savoir si l'on a affaire à une entreprise compétente.

Qualité de la direction : il se peut que tous les critères ci-dessus soient bons, mais que l'entreprise soit mal dirigée. Certains dirigeants sont de bons organisateurs mais de mauvais stratèges : celui qui est capable de concevoir une voiture de course n'est pas nécessairement le meilleur pilote de course. L'entreprise bien organisée a ce qu'il faut pour prendre de bonnes décisions, *encore faut-il qu'elle les prenne*.

Il existe des directions timorées, peu imaginatives, qui ne sont pas orientées vers l'action mais vers la paix du comité de direction (voir page 566). Il existe aussi des entreprises mal organisées qui ne s'en sortent pourtant pas mal, le pilote tirant bon parti d'une machine médiocre dont son « coup d'œil » compense à peu près les défauts.

Certaines entreprises ont un bon système d'information même si elles ne sont pas fortement informatisées¹⁴ : un petit carnet est parfois plus précieux qu'un ordinateur¹⁵.

Satisfaction des clients : l'évaluation de la qualité des produits doit être confortée par une enquête auprès d'un échantillon de clients ; il faudra vérifier, outre l'adéquation du produit aux besoins, la qualité du service commercial et du service d'après-vente, et poser des questions permettant d'évaluer la solidité du positionnement de l'entreprise face à la concurrence.

Engagements financiers : l'entreprise la plus efficace peut être « plombée » par un endettement trop lourd. Il faut évaluer les engagements financiers, y compris les engagements hors bilan, pour voir s'ils ne sont pas trop pesants en regard du cash-flow prévisible.

Satisfaction des créanciers et des actionnaires : la « crédibilité » de l'entreprise, clé de l'obtention des moyens financiers, s'évalue selon la confiance que lui accordent ses créanciers et actionnaires.

* *

Pour évaluer une entreprise il convient ainsi, avant tout recours à la comptabilité, de suivre la check-list suivante :

- 1) évaluer les trois critères principaux :
 - qualité de l'organisation ;
 - qualité du marketing ;
 - maîtrise de la fonction de production.
- 2) conforter l'analyse par l'évaluation de quatre critères secondaires mais importants :
 - qualité du système d'information ;
 - qualité des produits ;
 - qualité des compétences ;
 - qualité de la direction.
- 3) recouper les informations précédentes en évaluant :
 - la satisfaction des clients ;
 - les engagements financiers ;
 - la satisfaction des créanciers et des actionnaires.

14. C'est le cas des entreprises qui font un métier éventuellement compliqué mais selon une structure simple (réseau répétitif construit en nids d'abeille par réplication d'une même structure de base, comme celui d'une chaîne de magasins à grande surface). Les entreprises qui doivent synchroniser des activités diverses auraient par contre du mal à se passer d'un système d'information.

15. « L'illustre Galuchat était justement célèbre pour ses travaux sur les chaînes de montage. L'usine organisée suivant ses principes avait été fréquemment imitée, et était encore visitée par maintes délégations françaises et étrangères. La vente des systèmes de tapis roulants dits "chaînes Galuchat" avait rapporté gros à l'entreprise. Cependant, ses idées sur la gestion étaient singulières. Par exemple, il avait des paramètres de contrôle personnels, faisait faire de temps en temps quelques mesures, griffonnait parfois des règles de trois sur de petits bouts de papier » (« B.I.D.U.L.E. », 01-Informatique, décembre 76-janvier 77).

Après avoir parcouru cette check-list, il sera possible d'interpréter les données comptables, d'évaluer les anticipations et les risques, enfin de calculer la valeur de l'entreprise qui sera nécessairement présentée sous la forme d'une fourchette pour tenir compte des incertitudes.

15.2.2 Économie et urbanisation

Quel est le service que le système d'information rend à l'entreprise? cherchons, pour répondre à cette question, le meilleur compromis entre exactitude et brièveté: le système d'information, c'est le *langage de l'entreprise* (le langage qui se parle dans l'entreprise, et non pas le langage de programmation).

Les référentiels définissent les concepts selon lesquels elle décrit ses clients, ses produits, son organisation et ses procédures. Ce langage est en outre *outillé par l'automate programmable* qui assiste le travail mental de l'utilisateur en lui fournissant des moyens de classement, tri et traitement des données.

L'informatique équipe aujourd'hui la totalité du processus de conception et de production. Elle traite et classe non seulement des données structurées, mais aussi des textes en langage naturel, le courrier, la documentation, bref tout ce qui s'écrit dans l'entreprise. *Seuls lui échappent les textes effaçables*, les conversations orales et les notes manuscrites. La gestion des habilitations est donc devenue indispensable pour garantir la confidentialité: elle l'outille aussi.

Lorsque le système d'information est bien conçu, l'entreprise bénéficie d'un langage de qualité pour éclairer son positionnement, faire fonctionner ses procédures, évaluer son action. Elle maîtrise la communication multimédia entre ses agents et avec ses clients, ainsi que l'interopérabilité avec les systèmes d'information de ses partenaires. Alors le système d'information *élucide* l'entreprise, l'éclaire de telle sorte qu'elle rayonne l'information nécessaire aux acteurs.

Le système d'information a été souvent défini non par cette finalité, mais par les *moyens* que sont les ordinateurs, les programmes et les réseaux. On a cru que le système d'information, c'était de l'informatique. Mais la conception du langage de l'entreprise relève d'abord de la responsabilité des utilisateurs du système d'information.

Toute réflexion sur la qualité du langage pose des questions philosophiques délicates; par ailleurs, quand on structure et outille un langage, on définit une organisation. Il en résulte des réticences, des blocages à la fois intellectuels et sociologiques. La trivialité du *business is business* ne suffit plus pour y voir clair. Nos entreprises avancent donc lentement et maladroitement, comme quelqu'un qui marche à reculons.

* *

Le système d'information est pour l'entreprise un *actif*, un équipement, même s'il est d'un type particulier. Cependant les conventions comptables et fiscales masquent cette réalité. Les dépenses de développement étant considérées comme des frais d'exploitation, le système d'information n'apparaît

pas au bilan. L'AFAI (Association Française des Auditeurs Informatiques) est en train de construire la nomenclature qui permettra de connaître la fonction de coût du système d'information, mais jusqu'ici elle n'existait pas.

Les contraintes fiscales ne sont pas seules responsables. Certaines entreprises se laissent surprendre : leur budget informatique étouffe sous l'accumulation des dépenses de maintenance et sous le coût d'usage du réseau de PC.

Un projet sera d'autant plus visible, d'autant plus glorieux qu'il est plus coûteux, et non qu'il est plus utile. J'ai vu une entreprise payer deux millions d'euros pour une application qu'il aurait été possible de produire pour 100 000 € avec un progiciel de groupware : 100 000 €, c'était trop modeste pour apparaître dans le grain de la photo alors que les deux millions, étant visibles, se prêtaient à la discussion.

Plus un projet est gros, plus son risque d'échec est élevé. D'après le Standish Group un quart des grands projets informatiques échouent totalement (voir page 453). Nous ne supporterions pas un tel taux d'échec dans le BTP ou l'industrie : la maîtrise du système d'information manque de maturité.

* *

Mesurer la rentabilité d'un système d'information, cela équivaut à mesurer la rentabilité d'un langage de qualité ou d'une organisation correcte. Or si tout le monde sait qu'une entreprise mal organisée est inefficace, on n'applique pas pour autant le calcul de rentabilité à un effort d'organisation.

Posons cependant quelques repères. Aucun transporteur aérien ne peut vivre aujourd'hui sans système de réservation, sans segmentation tarifaire, sans supervision ; aucun opérateur télécoms ne peut vivre sans commutation électronique, sans système de facturation ; aucune banque, aucune compagnie d'assurance ne peuvent vivre sans une gestion informatique des comptes des clients. Dans ces entreprises, la rentabilité du système d'information est proprement infinie.

Le calcul économique indique cependant (voir page 537) qu'il existe pour chaque entreprise un taux d'informatisation optimal ainsi qu'une définition optimale du portefeuille applicatif - mais quel que soit son intérêt ce calcul théorique ne se prête pas à l'application pratique. L'entreprise ne peut approcher l'optimum que par tâtonnement, en suivant quelques règles simples qui se résument en trois mots clés que nous avons souvent répétés : pertinence, sobriété, cohérence (voir page 255).

La *pertinence*, c'est l'adéquation aux besoins des métiers. Elle s'acquiert par l'écoute des praticiens et par le *benchmark* auprès d'entreprises analogues, ainsi que par l'évaluation des applications en cours d'exploitation. Il s'agit d'une démarche expérimentale, qui procède donc au rebours du dogmatisme que l'on rencontre parfois à la direction générale.

La *sobriété* est le complément nécessaire de l'écoute. Si l'on suivait à la lettre la demande des utilisateurs on produirait des systèmes peu efficaces. Il faut prioriser, élaguer, simplifier le plus possible. C'est en partie parce que nos systèmes d'information sont trop lourds qu'il faut si souvent les remettre en chantier.

La plupart des systèmes d'information sont soumis à une entropie qui dégrade leur qualité sémantique. La *cohérence* s'obtient en articulant les applications au référentiel et en plaçant les données au centre du système d'information : c'est le principe même des ERP.

Ces trois critères illustrent la responsabilité de la maîtrise d'ouvrage. Les échecs informatiques ont *tous* été provoqués par des maîtrises d'ouvrage versatiles, incapables de définir leurs priorités, traversées par des conflits « politiques » au sens de la « politique d'entreprise », incapables d'équilibrer leur relation avec la maîtrise d'œuvre. Elles ont manqué de professionnalisme.

La technique permet aujourd'hui beaucoup de choses, à condition que l'entreprise respecte la compétence des informaticiens et sache les écouter. La modélisation du système d'information est délicate, mais là aussi méthodes et compétences existent. Le plus grand obstacle est ailleurs : on dirait que dans nos entreprises la lutte des classes a été remplacée par la *lutte des castes*. Le corporatisme, la défense des plates-bandes, l'entrelacement des réseaux d'influence et d'allégeance prennent trop souvent le pas sur la mission de l'entreprise.

Ce n'est pas une fatalité. Nos entreprises traversent une crise d'adaptation, une crise d'assimilation des TIC et de compréhension des enjeux du système d'information. De toute crise, on peut sortir.

15.2.3 Budget de l'informatique

Le réseau des postes de travail (PC, réseaux locaux, serveurs, routeurs, WAN, logiciels, support utilisateurs, maintenance), dont on parle rarement lors des discussions budgétaires, peut représenter plus de la moitié du coût d'un système d'information (voir page 529). Le reste se partage entre les dépenses consacrées aux serveurs centraux d'une part, celles consacrées aux progiciels et logiciels d'autre part.

La maîtrise économique du système d'information suppose que l'on élucide la dynamique de ses coûts : les coûts d'investissement d'aujourd'hui suscitent des coûts de maintenance et d'exploitation durant les années suivantes (voir page 531). Le stock que représente le système d'information est ainsi alimenté par une accumulation d'investissements et diminué par l'obsolescence.

La discussion budgétaire se focalise le plus souvent sur le coût de l'investissement informatique (les « projets »). Elle ignore alors le coût de la maintenance ainsi que le coût de la maîtrise d'ouvrage. Alors que les entreprises évaluent le coût des ressources humaines externes au prix du marché, le coût des ressources internes est évalué selon un prix conventionnel inférieur au prix du marché. Il peut donc arriver qu'il soit sous-évalué, et que l'on néglige le coût de la désorganisation que provoque la mobilisation des experts du terrain par la maîtrise d'ouvrage.

* *

La connaissance des coûts vise des finalités diverses : s'assurer que l'on respecte une enveloppe financière ; comparer diverses solutions entre elles ;

vérifier que l'on n'est pas en train de « dérapier ». Des évaluations en homme-jour peuvent à la rigueur suffire pour traiter ce dernier point, mais il faut impérativement mesurer des coûts en euros si l'on veut obéir à l'une des deux autres finalités.

Souvent la DSI ne fournit pas l'information sur les coûts : elle dispose certes de toute l'information détaillée, mais elle ne la traite pas de façon convenable soit parce qu'elle est sous-informatisée¹⁶, soit parce qu'elle en élabore une présentation purement comptable et non économique. Parfois le DSI craint que le système d'information ne semble trop onéreux à l'entreprise, car elle ne sait pas en évaluer la rentabilité : alors il préfère qu'elle ignore son coût.

La codification comptable des dépenses est souvent malencontreuse et gêne la compréhension de l'économie du système d'information. Le contrôleur de gestion et le directeur financier raisonnent alors selon une classification qui masque l'économie du système d'information. Il faut que l'entreprise sache utiliser deux langages : celui de la comptabilité, nécessaire pour répondre à la réglementation fiscale, et celui de l'économie qui seul permet de comprendre l'entreprise.

Examiner le *cycle de vie* d'un logiciel permet de n'oublier aucun poste de dépense : cela implique en effet d'évaluer les coûts de maintenance et la durée de vie. Pour estimer cette dernière il faut considérer que l'obsolescence dépend d'une part de l'usure du logiciel, elle-même fonction du nombre de modifications qu'on lui apporte (un logiciel se dégrade chaque fois qu'il est modifié), d'autre part de l'évolution du métier dont les processus peuvent dériver par rapport à ce qui avait été modélisé (réglementation, adaptation à de nouveaux marchés etc.). Souvent l'évolution du métier obligera à retoucher le logiciel, ce qui accélérera son usure.

* *

La préparation du budget du système d'information part (ou devrait partir) des informations suivantes :

- un cadre (plan d'urbanisme, schéma d'évolution etc.) donnant une visibilité pluriannuelle sur l'évolution du système d'information, et mis à jour chaque année ;
- la liste des projets en cours qu'il est nécessaire de poursuivre ;
- les nouvelles demandes des maîtrises d'ouvrage (les demandes sont-elles prioritaires ou non ? le coût de chacune a-t-il été évalué ainsi que leur rentabilité ? les demandes nouvelles sont-elles classées en « TMA », avec l'ambiguïté que cela comporte, ou en « projet nouveau » ?) ;
- les demandes de la DSI : maintenance, redimensionnement ou rénovation de la plate-forme technique (réseaux, machines, logiciels) ;
- l'état de la trésorerie : reports de budget disponibles, contraintes financières de l'entreprise.

16. C'est paradoxal, mais c'est souvent le cas.

La qualité de l'arbitrage budgétaire dépend de la méthode suivie : a-t-il été convenablement préparé (clarté de la gestion de portefeuille, de la trésorerie), les parties prenantes sont-elles convenablement associées à la décision (directeur général, DSI, MOAS, MOAD, MOAO) ?

Ensuite la portée de la décision budgétaire dépend de son appropriation par l'entreprise : s'agit-il *vraiment* d'une décision qui sera effectivement appliquée, ou pourra-t-elle être remise en question ou même ignorée en pratique ? la vision pluriannuelle du système d'information sera-t-elle mise à jour à partir de cette décision ?

On va par exemple au printemps définir le budget de l'année prochaine sous forme d'une « lettre plafond » accompagnée d'un argumentaire expliquant les raisons du cadrage. Cette lettre plafond permet à chacun de savoir où va l'entreprise : il faut qu'elle soit signée par le directeur général pour éviter toute contestation.

Puis les maîtrises d'ouvrage font des demandes argumentées par application ou ensemble d'applications. Il s'ensuit des arbitrages à divers niveaux, jusqu'à ce que la prévision de dépense globale ait pu entrer dans le cadre fixé par la lettre plafond. Le point le plus important est de définir, puis gérer la masse financière sur laquelle on va pouvoir jouer une fois payés le fonctionnement et la poursuite des projets engagés (masse que l'on appellera « enveloppe réservée », « non récurrent », « budget libre » etc., et qui détermine le montant à consacrer aux nouveaux projets).

Le lancement des nouveaux projets focalise souvent la discussion. Cependant un projet pluriannuel qui n'est pas terminé peut être soit reconduit, soit remis en cause en tout ou en partie : la discussion budgétaire donne l'occasion de réviser les décisions antérieures.

La qualité de la procédure suivie pour faire émerger les meilleurs projets est un enjeu important ; les directions qui maîtrisent l'art de bien présenter un dossier, de bien plaider leur cause, disposent d'un avantage tactique d'autant plus fort que lorsqu'on élabore le budget (au début de l'été de l'année $A-1$) on ne peut évaluer précisément ni le coût du projet, ni ce qu'il va rapporter. Pour que la préparation du budget aboutisse à des conclusions raisonnables, il faut qu'elle soit encadrée par une procédure qui mette en évidence les sujets les plus sensibles.

* *

Il importe de savoir *qui* maîtrise l'exécution du budget. A-t-il été attribué à la DSI (sous la forme d'une « enveloppe informatique ») ou à chaque direction, responsable du coût de son système d'information ?

Si le budget a été attribué à la DSI, quels sont les moyens dont dispose une direction pour faire prévaloir ses objectifs ? Lui sera-t-il possible de mettre la DSI en concurrence avec des fournisseurs extérieurs à l'entreprise ? d'exercer des sanctions si la fourniture n'est pas conforme en termes de délais ou de qualité ?

Qui dirige le projet : un directeur de projet issu de la maîtrise d'ouvrage, ou un chef de projet MOE ? Ce directeur de projet dispose-t-il réellement de l'autorité qui permet de le diriger - et en particulier, vérifie-t-il le service

fait par les fournisseurs, autorise-t-il le paiement de leurs factures - ou bien est-il une potiche, responsable d'une fonction qu'il n'a pas les moyens de remplir, coupable désigné d'avance en cas d'échec?

Chaque projet est suivi par divers comités (comité d'avancement, comité de pilotage, comité directeur). Leurs décisions sont-elles suivies d'effet, ou les réunions se réduisent-elles à une liturgie?

Les comités sont informés par les rapports périodiques qui leur sont communiqués. Ces rapports donnent-ils des indications sur la quantité de travail consommée? le taux de réalisation des livrables? le pourcentage du budget dépensé? Donnent-ils une visibilité sur les retards? sur les dépassements? comportent-ils des alarmes? Sont-ils remplis avec sérieux? (un reporting dont le format est convenable, mais qui serait mal rempli, ne sert à rien¹⁷).

L'entreprise a-t-elle une vue complète sur le coût de son système d'information? est-elle en mesure de distinguer les coûts de la maîtrise d'œuvre et ceux de la maîtrise d'ouvrage, entre les coûts externes et internes, les dépenses d'investissement, de maintenance et d'exploitation? d'établir un rapport entre le coût et l'efficacité du système d'information? d'évaluer sa rentabilité?

15.3 Calcul économique

Pour modéliser l'économie de l'informatique du point de vue de l'entreprise qui l'utilise, il faut d'abord examiner la fonction de production d'un système d'information. Celui-ci comporte trois couches : la plate-forme physique, la plate-forme logicielle, les programmes.

Tout système d'information repose sur une plate-forme qui comporte les serveurs centraux (*mainframes*) avec leurs ressources de puissance (processeurs) et de mémoire (disques, RAM), puis le réseau de données avec les circuits et routeurs, enfin les réseaux locaux de PC avec les serveurs locaux et les postes de travail.

À cette plate-forme physique est associée une plate-forme logicielle (systèmes d'exploitation, langages de programmation, ateliers de génie logiciel, progiciels et interfaces) qui la rend capable d'accueillir les programmes informatiques que met en œuvre le système d'information.

Les plates-formes physique et logicielle, les programmes, résultent d'un investissement et constituent donc un capital fixe qu'il faudra par la suite exploiter, maintenir et faire évoluer tout en apportant aux utilisateurs les services d'accompagnement nécessaires à sa bonne utilisation (soutien de proximité, dépannage, *help desk* etc.).

Pour établir la fonction de coût, il faudra ajouter au coût de ces divers éléments ceux de la maîtrise d'ouvrage (conception fonctionnelle, recette, formation des utilisateurs, déploiement etc.).

17. On observe que la qualité des rapports tend à se dégrader en cours de projet : on démarre avec un bon format de reporting, puis on le remplit de plus en plus mal et si l'on n'y prend garde il devient inefficace : alors le projet est conduit à l'aveuglette...

Une fois que l'entreprise a déterminé les fonctionnalités dont elle souhaite équiper ses processus de production on peut évaluer le volume des programmes (qu'il soit mesuré par le nombre de lignes de code source, le nombre de points de fonction ou par tout autre indicateur de volume) qu'il faut produire pour réaliser ces fonctionnalités, puis estimer la durée et le coût du développement ainsi que le nombre des programmeurs nécessaires. Ici on dispose de « règles de pouce » qui fournissent des indications certes imprécises, mais utiles pour éclairer la décision comme le pilotage des projets¹⁸.

Il est difficile d'évaluer la rentabilité du système d'information car ce qu'il produit est souvent impossible à quantifier (il serait, de même, difficile d'évaluer la rentabilité d'un effort d'organisation : cela ne veut pas dire que l'organisation soit inutile!). Il en résulte que des directions générales, dépourvues des critères qui permettraient une évaluation quantitative, s'en remettent à certaines règles simples (maintenir le budget informatique constant, maintenir constante la part des nouveaux développements dans le budget etc.). Il est utile d'examiner ce que peuvent être les conséquences de ces règles simples lorsque leur application est poussée au delà du raisonnable : nous avons construit un modèle de simulation qui fera apparaître d'éventuels effets pervers.

La critique des règles simples conduit naturellement à se demander quel est le « bon » niveau du système d'information, le degré d'informatisation que l'on puisse juger optimal. Pour y répondre, il faut poser des hypothèses sur la contribution du système d'information à la production de l'entreprise. Un programme que l'on développe dans l'année A sera utilisé durant les années suivantes : les programmes constituent donc un *stock*. Le choix du niveau optimal d'informatisation relève d'un arbitrage intertemporel que l'on pourra modéliser si l'on précise la fonction objectif de l'entreprise.

Une fois posées ces hypothèses, on pourra en déduire la dynamique optimale de l'informatisation, c'est-à-dire la chronique des dépenses informatiques qui permet de maximiser la fonction objectif une fois maîtrisés les risques associés à la maîtrise des délais et des coûts de réalisation de chaque programme.

15.3.1 Rentabilité d'un ensemble de projets

À tout projet on peut associer l'anticipation d'un flux de trésorerie disponible¹⁹. Comme toute anticipation, elle est sujette à l'incertitude et il convient d'en tenir compte.

Le flux de trésorerie F_t disponible lors de l'année t est, à la variation du besoin de fonds de roulement près, l'écart annuel entre les recettes et les

18. On trouve une présentation de ces règles dans Printz [165].

19. Cette anticipation prend la forme d'une série chronologique, la valeur F_t affectée à l'année t étant égale au cash-flow (solde des flux de trésorerie) que procure le projet, diminué de la variation du besoin de fonds de roulement qu'il suscite. Ces évaluations sont faites par différence entre deux scénarios : un scénario tendanciel, dans lequel le projet ne serait pas réalisé, et le scénario qui résulterait de la réalisation du projet.

dépenses associées au projet. Si l'on considère par exemple un projet qui demande une dépense d'investissement \mathcal{K} lors de l'année 0, puis qui rapporte durant chacune des années suivantes et jusqu'à l'infini le résultat brut \mathcal{R} , la série F_t s'écrira : $F_0 = -\mathcal{K}$, $F_1 = \mathcal{R}$, $F_2 = \mathcal{R}$, $F_3 = \mathcal{R}$ etc.

Nous appellerons « projet simple » un tel projet. La plupart des projets ne sont pas « simples » : les valeurs F_t évoluent dans le temps et la durée de vie du produit que le projet élabore se limite à quelques années. Cependant nous allons voir qu'il est possible d'associer à chaque projet un « projet simple » qui le représentera utilement.

* *

On calcule, à partir du flux de trésorerie que suscite un projet, deux indicateurs synthétiques : la « valeur actuelle nette » (VAN) et le « taux de rentabilité interne » (TRI, en anglais ROI pour *Return on Investment*²⁰). Nous noterons V la VAN et τ le TRI d'un projet.

La VAN est la somme actualisée de la variation de trésorerie disponible F_t ; le taux d'actualisation r à retenir est égal au taux d'intérêt du marché monétaire i , augmenté pour tenir compte de l'incertitude de la prime de risque π : $r = i + \pi$. Le choix du taux d'intérêt et de la prime de risque est fait par la direction générale en s'appuyant sur l'expertise de la direction financière et il doit être révisé chaque année. La prime de risque peut en outre varier d'un projet à l'autre selon la « classe de risque » à laquelle appartient chaque projet.

La VAN d'un projet s'écrit alors, en sommant sur la durée de vie d du produit :

$$V = \sum_0^d \frac{F_t}{(1+r)^t}$$

Le TRI τ est le taux d'actualisation pour lequel la VAN serait nulle. Il est donc solution de l'équation :

$$\sum_0^d \frac{F_t}{(1+\tau)^t} = 0$$

On doit rejeter immédiatement tout projet dont le TRI serait inférieur au taux d'actualisation car sa VAN serait négative.

Si l'on considère un projet simple qui coûterait \mathcal{K} la première année, puis rapporterait indéfiniment le résultat brut \mathcal{R} lors chacune des années suivantes, l'expression de la VAN et du TRI est elle-même simple ; on trouve :

$$V = \frac{\mathcal{R}}{r} - \mathcal{K} \quad ; \quad \tau = \frac{\mathcal{R}}{\mathcal{K}}$$

La deuxième équation permet de comprendre ce qu'est le TRI : un projet simple est analogue à un placement d'un montant \mathcal{K} au taux d'intérêt τ .

20. Certains utilisent l'acronyme ROI pour désigner le délai du retour sur investissement, c'est-à-dire le délai au bout duquel le cumul du flux de trésorerie s'annule, le flux des recettes induites par le projet ayant exactement compensé son coût. On entend dire par exemple « le ROI est de six mois ».

Considérons un projet donc nous connaissons la VAN V et le TRI τ . On peut lui associer un projet simple qui aura la même VAN et le même TRI que lui, et qui lui sera donc équivalent à cet égard.

À partir des équations ci-dessus, un petit calcul fournit les valeurs de \mathcal{K} et de \mathcal{R} relatives à ce « projet simple » :

$$\mathcal{K} = \frac{rV}{\tau - r} \quad ; \quad \mathcal{R} = \frac{rV\tau}{\tau - r}$$

À chaque projet caractérisé par sa VAN et son TRI, ces relations permettent d'associer les valeurs \mathcal{K} et \mathcal{R} correspondant au projet simple qui lui serait équivalent en ce sens qu'il aurait la même VAN et le même TRI.

Dans le plan $(\mathcal{K}, \mathcal{R})$ un projet sera ainsi représenté par un point \mathcal{P} , la pente du segment $O\mathcal{P}$ étant égale à son TRI puisque $\tau = \mathcal{R}/\mathcal{K}$ (figure 15.1).

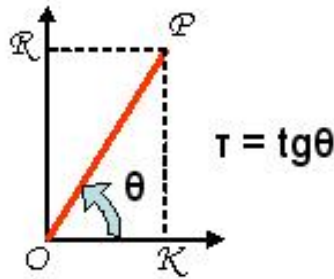


FIG. 15.1 – Point représentatif d'un projet

Classons alors les projets dans l'ordre des TRI décroissants en enchaînant l'un à l'autre des segments de droite représentant chacun un projet. On obtient une courbe polygonale dont la concavité est tournée vers le bas (figure 15.2).

Supposons que l'on décide de réaliser les projets dont le TRI est le plus élevé en bornant l'effort d'investissement par un montant fixé à l'avance. La dépense initiale totale sera la somme K des coûts de ces projets, le résultat brut annuel total R sera la somme des résultats bruts ; la rentabilité de l'investissement total ainsi réalisé est le rapport R/K de ces deux totaux. Ce rapport est le taux de rentabilité maximal que l'entreprise puisse obtenir pour l'effort K .

Le raisonnement ci-dessus appelle quelques remarques et précautions complémentaires :

1) Le budget annuel nécessaire à la réalisation de ces projets ne sera pas exactement égal à la somme K , puisque nous avons considéré pour évaluer les coûts \mathcal{K} non les projets eux-mêmes, mais des « projets simples » qui leur sont équivalents. Néanmoins K donne une mesure de l'effort d'investissement nécessaire. On peut d'ailleurs en partant de la liste des projets et, pour chacun d'entre eux, de la chronique des investissements, évaluer l'effort annuel correspondant.

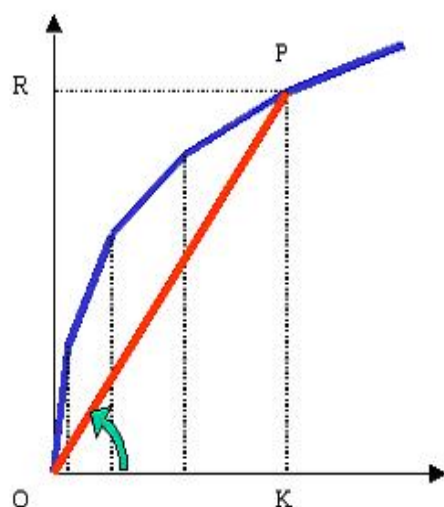


FIG. 15.2 – Rentabilité d'un ensemble de projets

2) Le plus souvent, le montant K fixé à l'avance ne correspondra pas exactement au financement de n projets car il ne « tombe pas juste ». Il revient au directeur général de décider s'il révisé K à la hausse (pour pouvoir financer un projet de plus) ou à la baisse, afin qu'il corresponde exactement au financement de n projets.

3) Les évaluations qui fondent le classement sont toujours entachées d'incertitude et par ailleurs le choix du taux d'actualisation r comporte une part d'arbitraire ; toutefois comme elles résument les informations que l'entreprise détient au moment de la décision *il n'est pas possible de faire mieux* pour préparer celle-ci. L'entreprise ne pourra améliorer ces évaluations que si elle capitalise son expérience par l'analyse rétrospective des projets réalisés et de leurs résultats, ce qui demandera un effort persévérant sur plusieurs années.

4) Le raisonnement ci-dessus suppose que l'on a déjà pris en compte les synergies entre projets. Si la synergie entre les projets P_1 et P_2 est élevée (en les réalisant tous les deux, on obtiendrait un résultat significativement supérieur à la somme des résultats qu'ils procureraient séparément), il faut les regrouper en un même projet ; si leur synergie est fortement négative (ils utilisent tous deux une même ressource rare, ou bien leur réalisation conjointe brouillerait le positionnement de l'entreprise), il ne faut retenir que l'un des deux. *Ces décisions doivent avoir été prises avant la présentation de la liste des projets au directeur général.*

6) Le calcul de rentabilité ignore les aspects d'un projet qui sont difficiles à « chiffrer », par exemple son incidence sur l'évolution du positionnement de l'entreprise. C'est pourquoi le classement par ordre de rentabilité décroissante doit être soumis à l'arbitrage du directeur général qui sera libre, s'il le juge utile, de modifier l'ordre des priorités, de « tuer » tel projet que

l'on avait jugé rentable ou de « repêcher » tel autre projet qui avait été mal classé. Il faut que les experts acceptent la valeur ajoutée qu'apporte le dirigeant, qui est le mieux placé pour avoir une vue d'ensemble des enjeux stratégiques.

7) Certains estiment que l'importance d'un projet se mesure non selon son TRI, mais selon sa VAN : ils classent donc les projets non dans l'ordre des TRI décroissants mais dans l'ordre des VAN décroissantes. Cependant comme la VAN est fonction croissante à la fois du TRI et de la taille du projet son montant résulte du jeu conjugué de ces deux paramètres. Nous préférons pour notre part laisser au directeur général le soin d'apprécier l'importance relative des projets et lui présenter un classement qui, fondé sur le seul critère du TRI, maximise la rentabilité (anticipée) de l'investissement global.

15.3.2 Durée de vie d'un logiciel

La durée de vie d'un logiciel résulte de deux phénomènes : d'une part le coût de maintenance s'accroît avec l'âge du logiciel ; d'autre part un écart se creuse, en raison de l'évolution de l'entreprise, entre les fonctionnalités qu'il offre aux utilisateurs et les besoins de ceux-ci. Nous ne considérerons ici que le premier de ces deux phénomènes, ce qui revient à supposer à titre d'exercice que l'entreprise n'évolue pas.

Notons n la taille du logiciel, qu'elle soit mesurée par le nombre de lignes de code, de points de fonction ou par toute autre unité. La complexité du logiciel est $f(n)$, fonction croissante de sa taille (nous n'aurons pas besoin ici de spécifier davantage cette fonction). Notons n_0 la taille initiale du logiciel. Nous supposons le coût initial C_0 de production proportionnel à sa complexité : $C_0 = kf(n_0)$.

Une fois le logiciel en service, des travaux de maintenance seront nécessaires pour l'adapter aux modifications de l'environnement technique (architecture de la plate-forme, nouvelles versions des systèmes d'exploitation etc. ; nous ne considérons pas ici la « maintenance évolutive » qui consiste à introduire de nouvelles fonctionnalités et doit en fait être considérée comme un nouveau développement). Les travaux de maintenance conduisent à écrire de nouvelles lignes de code qui s'ajoutent aux lignes initiales. La taille du logiciel croît donc au fur et à mesure de son vieillissement²¹.

Supposons que le coût annuel m_t de la maintenance soit celui de la croissance de la complexité du logiciel : $m_t = k\Delta f(n_t)$. Supposons par ailleurs que ce coût soit égal au coût cumulé du logiciel (y compris la somme des coûts de maintenance antérieurs) multiplié par un coefficient p : $m_t = pkf(n_t)$. Il résulte de ces deux égalités que $\Delta f(n_t)/f(n_t) = p$, d'où l'évolution de la complexité $f(n_t) = f(n_0)e^{pt}$.

21. Parfois, les nouvelles lignes de code *remplacent* les lignes initiales (et donc la taille du logiciel ne croît pas). Mais leur introduction se paie souvent par une complication du logiciel, voire par une perte de cohérence, tant il est difficile de vérifier celle-ci lors d'une correction ; ce phénomène équivaut qualitativement à une croissance de la taille du logiciel.

Le coût annuel de la maintenance est alors à partir de l'année 1 :

$$m_t = pC_0 e^{pt}$$

Si l'on suppose que la totalité du coût initial C_0 est dépensée lors de l'année 0 et si $p = 15\%$ (valeur que l'on retient parfois), la série des coûts annuels a l'allure indiquée par la figure 15.3.

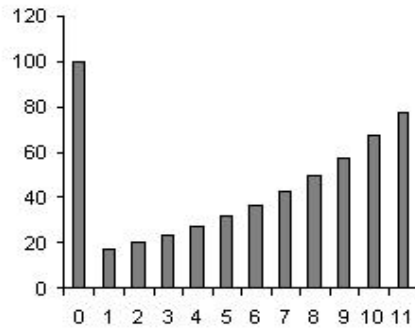


FIG. 15.3 – Evolution du coût annuel

Comme le coût de la maintenance augmente exponentiellement il sera préférable, au bout d'un certain délai, de refaire le logiciel à neuf : nous appellerons ce délai « durée de vie du logiciel ». Le but du calcul ci-dessous est d'évaluer la *durée de vie optimale*, qui minimise le coût actualisé du logiciel.

* *

Supposons que l'entreprise retienne le taux d'actualisation r et que la durée de vie du logiciel soit d .

À la fin de la durée de vie, le logiciel est entièrement refait pour le même coût C_0 , puis s'ensuit un nouveau cycle de vie.

Le coût actualisé du logiciel sur un cycle de vie est :

$$A = C_0 \left(1 + p \int_0^d e^{(p-r)t} dt \right)$$

Soit, si $p \neq r$,

$$A = C_0 \left[1 + p(e^{(p-r)d} - 1)/(p - r) \right]$$

Par ailleurs l'annuité équivalente à A est a telle que :

$$A = a \int_0^d e^{-rt} dt = a(1 - e^{-rd})/r, \text{ d'où}$$

$$a = rC_0 \frac{pe^{(p-r)d} - r}{(1 - e^{-rd})(p - r)}$$

La durée de vie qui minimise a est alors d^* telle que :

$$p(p-r) - p^2 e^{-rd^*} + r^2 e^{-pd^*} = 0$$

Il n'est pas possible de trouver la solution analytique de cette équation mais on peut la résoudre par le calcul numérique. Supposons que $p = 15\%$ et que $r = 9\%$ (ce sont là des valeurs que l'on peut rencontrer dans la pratique). On trouve que $d^* = 7,23$ ans et que l'annuité équivalente minimale a^* est égale à 44,36 % de C_0 .

La figure 15.4 montre comment a varie avec la durée de vie du logiciel. Le minimum de a est plat : pour des durées de vie de 6 à 9 ans, l'écart entre a et l'optimum a^* est inférieur à 2 %. Par contre le coût sera sensiblement plus élevé si la durée de vie sort de cet intervalle.

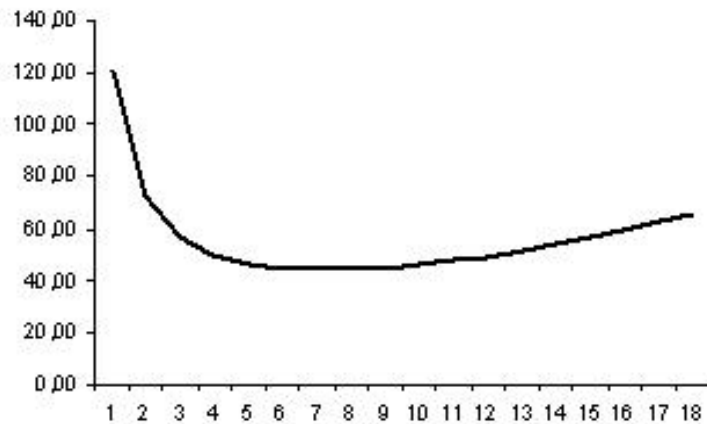


FIG. 15.4 – Annuité équivalente en fonction de la durée de vie

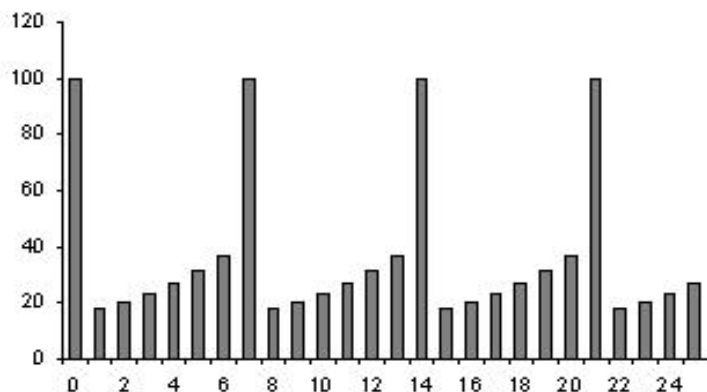
Le logiciel sera refait à neuf toutes les d^* années ; la chronique des dépenses annuelles aura non l'allure que l'on voit sur la figure 15.3, mais celle indiquée par la figure 15.5, où l'on a retenu une durée de vie de 7 ans.

On observe que le coût de maintenance est, à la fin de la durée de vie, égal à 36,89 % de C_0 : ce résultat, conjugué au fait que nous avons choisi pour p et r des valeurs réalistes, correspond à une règle de pouce d'usage courant : « il faut refaire un logiciel quand le coût de maintenance annuel atteint le tiers du coût de réfection ».

* *

Supposons maintenant que le coût de réfection du logiciel diminue dans le temps au rythme annuel s , par exemple en raison des progrès des langages de programmation.

Notons donc A_c le coût du cycle de vie de rang c : $A_c = A e^{-scd}$, où A est le coût du premier cycle de vie et d la durée de vie. Le coût actualisé de

FIG. 15.5 - *Dépenses annuelles*

la succession des cycles de vie est :

$$S = \sum_{c=0}^{\infty} e^{-rcd} A_c = A / (1 - e^{-(r+s)d})$$

L'annuité équivalente à cette dépense est $a = rS$, d'où :

$$a = rC_0 \frac{pe^{(p-r)d} - r}{(1 - e^{-(r+s)d})(p - r)}$$

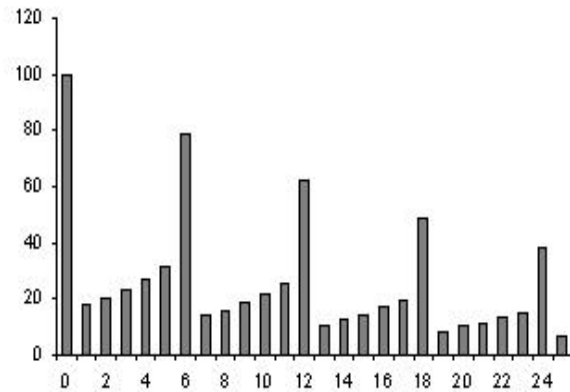
Si l'on suppose que $p = 15\%$, $r = 9\%$ et $s = 4\%$, on trouve que $d^* = 6,33$ ans et que a^* est égal à $34,58\%$ de C_0 .

La baisse tendancielle du coût de réfection a donc pour effet de réduire la durée de vie du logiciel ainsi que l'annuité équivalente. La chronique des dépenses annuelles a alors l'allure que fait apparaître la figure 15.6.

15.3.3 Coût des PC en réseau

L'évaluation ci-dessous résulte d'une étude de cas réalisée en 1997 à l'occasion de l'analyse du budget informatique d'une entreprise que nous appellerons PHARE. L'accès à de telles informations n'est pas aisé car pas plus que d'autres directeurs les DSI n'aiment que l'on regarde leurs comptes de près : en l'occurrence, la fenêtre qui s'était entrouverte s'est vite refermée mais nous avons eu le temps de voir certaines choses.

Les données ont été retraitées pour préserver l'anonymat de cette entreprise tout en conservant les proportions significatives. On n'a procédé à aucune ventilation des dépenses générales (encadrement etc.), dont une partie aurait pu être affectée au coût des PC en réseau. Il aurait fallu disposer d'une série sur plusieurs années pour renforcer la qualité des estimations, mais nous n'avons pas pu l'obtenir.

FIG. 15.6 – *Dépenses annuelles avec baisse du coût*

On a repéré dans les comptes toutes les dépenses immédiatement liées au réseau de PC : le coût des matériels, des logiciels, de la maintenance, du *help desk*, de l'administration et de la supervision du réseau etc. On n'a pas pris en compte le coût du support bureautique de proximité (on estime qu'il faut lui consacrer un emploi pour cinquante à cent salariés), ni le coût de la formation des utilisateurs. Le coût des matériels est comme il se doit un coût d'usage annuel (dépenses de location ou amortissement).

L'étude a fait ressortir des proportions qui peuvent surprendre. On ne doit pas tirer d'une monographie plus qu'elle ne peut donner : ce que l'on a observé sur PHARE est relatif à cette entreprise et à elle seule. Néanmoins le caractère surprenant des ordres de grandeur encourage à faire des études analogues dans les autres entreprises.

* *

PHARE est une entreprise de service qui possède plus de 1000 implantations en France. En 1997, 13 000 salariés étaient équipés de PC en réseau.

Le budget informatique annuel est de 76 millions d'euros. La part des PC en réseau dans le coût total de l'informatique est de 58 % (dont 35 % pour les PC et 23 % pour le réseau) : c'est la première des surprises apportées par cette étude. Lors des réunions du Comité Système d'Information de PHARE, l'essentiel des débats porte sur les grands projets alors que plus de la moitié du budget informatique est consommée par le réseau de PC dont on ne parle jamais : on fait comme si ce réseau allait de soi et fonctionnait tout seul.

Évaluer le coût du réseau de PC a l'avantage de nous inviter à poser solidement les pieds par terre en faisant percevoir les proportions qui prévalent dans le coût de l'informatique. Ces proportions rappellent la structure du coût d'un réseau de télécommunications : les parties « nobles » du réseau (commutation, transmission) ne représentent que le quart du coût total,

alors que la « boucle locale » (réseau de distribution assurant le raccordement des abonnés), qui recourt aux techniques les moins sophistiquées (génie civil, connectique) en représente les trois quarts.

* *

Les personnels consacrés aux PC en réseau (installation des réseaux et des PC, ingénieurs système réseau, ingénieurs système Micro et Serveur) représentent 34 % des effectifs de l'informatique et 31 % de ses frais de personnel. Les PC en réseau consomment par ailleurs 15 % des dépenses en prestataires externes. La dépense en personnel la plus importante est celle relative à l'installation et au support aux utilisateurs.

Le coût des logiciels consacrés au réseau de PC représente 86 % du coût d'acquisition des logiciels et 51 % de leur coût de maintenance. Le poste de coût le plus important est la maintenance des logiciels réseau.

Les dépenses consacrées au réseau et au PC représentent 71 % des dépenses totales de matériel et 59 % des dépenses de maintenance du matériel. Les serveurs et micros consomment l'essentiel de cette dépense.

Enfin, le coût annuel total des réseaux utilisés (Transpac, Transfix, Numéris etc.) est de 13 millions d'euros.

Au total, le coût annuel du réseau de PC est de 3 343 euros par PC. Le poste le plus important est le coût du matériel.

15.3.4 Règles de gestion et optimisation

Le but de cette section est de simuler la dynamique d'un système d'information lorsque l'entreprise détermine l'effort qu'elle consacre à celui-ci selon l'une ou l'autre des règles simples que l'on rencontre dans la pratique (Peaucelle [156]).

Les directions générales perçoivent en effet souvent l'informatique comme un poste de coût qu'il faut comprimer sans s'interroger plus avant sur une rentabilité dont elles croient l'évaluation impossible. Elles choisissent alors de suivre l'une ou l'autre de quelques règles simples : fixer le budget de l'informatique à un montant jugé raisonnable, qu'elles appellent « enveloppe » ; fixer le montant consacré aux nouveaux projets ; fixer la part des nouveaux projets dans le budget informatique etc. Mais certaines de ces règles sont susceptibles d'avoir des effets pervers²².

²². Le détail des calculs ci-dessous est publié à l'adresse www.volle.com/travaux/modelesi1.htm

Pour pouvoir les explorer, nous poserons les hypothèses suivantes :

- le système d'information comporte à la date t un stock d'applications dont le volume²³ est mesuré par S_t , le coût de maintenance et d'exploitation par $a_t S_t$ ²⁴ ;

- durant l'année t sont réalisées de nouvelles applications qui viendront s'ajouter aux anciennes et dont le coût de production est $b_t N_t$ (nous comptons parmi les nouvelles applications les évolutions des applications anciennes) ;

- le taux d'obsolescence est δ ($0 < \delta < 1$)²⁵ ;

- les coûts unitaires décroissent au taux annuel constant k à partir des valeurs initiales $a_0 = a$ et $b_0 = b$:

$$a_t = a(1 - k)^t ; \quad b_t = b(1 - k)^t$$

Le coût du système d'information à l'année t est alors :

$$C_t = a_t S_t + b_t N_t = (a S_t + b N_t)(1 - k)^t$$

Le stock d'applications en exploitation durant l'année $t + 1$ est :

$$S_{t+1} = S_t + \Delta S_t = S_t(1 - \delta) + N_t$$

Si l'on connaît le niveau initial S_0 et la série N_t , on peut à partir des relations précédentes calculer les séries S_t et C_t . Les deux équations ci-dessus permettent donc, une fois connus S_0 et N_t , de décrire entièrement la dynamique du coût du système d'information.

D'après Keen [99] pour une dépense de 1 \$ en développement on dépense pendant chaque année ultérieure 20 cents d'exploitation et 40 cents de maintenance. Nous retiendrons donc ici les hypothèses suivantes :

- le coût annuel de la maintenance est de 20 % du coût de réalisation (il se situe entre le taux initial de 15 % et le taux de 33 % au delà duquel l'application serait jugée obsolète),

- le coût annuel d'exploitation est de 30 % du coût de réalisation.

23. La mesure de ce « volume » serait en pratique délicate : on pourrait l'évaluer en comptant le nombre de points de fonction inclus dans les applications, le nombre de lignes de code, on pourrait utiliser diverses pondérations etc. La mesure n'est cependant ici, observons le, ni plus ni moins difficile que dans d'autres domaines : toute mesure de volume suppose d'agréger des choses différentes, ce qui ne peut se faire qu'en adoptant des conventions toujours discutables.

24. Le coût de maintenance d'une application croît avec le temps ; cela détermine sa durée de vie à besoins fonctionnels constants car il faut la refaire quand le coût de maintenance devient trop élevé par rapport au coût de réfection. Cependant si l'on considère l'ensemble du système d'information la dépense de maintenance, somme des coûts de maintenance d'applications d'âges divers, peut être considérée comme une fraction du coût de réalisation initial du stock d'applications. Pour simplifier notre modèle nous la supposons ici constante (on pourrait enrichir et compliquer le modèle en considérant la pyramide des âges du stock d'applications mais ses résultats n'en seraient pas sensiblement modifiés).

25. δ est l'inverse de la durée de vie d'une application (il est donc nul si l'on suppose la durée de vie infinie).

Si l'on normalise le coefficient b (coût unitaire de la réalisation) en lui donnant la valeur 1, le coefficient a (coût unitaire de l'exploitation et de la maintenance) est alors égal à 0,5.

* *

Nous allons examiner l'évolution du budget informatique sous diverses hypothèses sur le comportement de la direction générale et qui déterminent chacune la série N_t :

- valeur constante du flux des nouveaux développements ;
- part constante des nouveaux développements dans le budget ;
- budget constant.

Les entreprises qui les appliquent peuvent certes changer la règle quand ses conséquences deviennent déraisonnables, mais il faut qu'elles soient vigilantes : si une même règle est appliquée trop longtemps avant que l'entreprise ne réagisse les dégâts peuvent être considérables : réguler en appliquant une succession de règles explosives présente des risques qu'il faut savoir maîtriser.

Les simulations numériques sont réalisées à partir des hypothèses suivantes : $a = 0,5$; $b = 1$; $\delta = 20\%$ (la durée de vie est de cinq ans) ; $k = 20\%$. Pour calculer les valeurs limites, nous tirerons parti d'un théorème classique : « Si la série u_n est telle que $u_{n+1} = au_n + b$ avec $|a| < 1$, u_n converge vers la limite $u^* = b/(1 - a)$ ».

Coût constant des nouveaux développements

Supposons constant le budget consacré aux nouveaux développements : $b_t N_t = d$.

Le coût annuel C_t du système d'information tend alors vers C^* :

$$C^* = d \left[1 + \frac{a(1 - k)}{b[1 - (1 - \delta)(1 - k)]} \right]$$

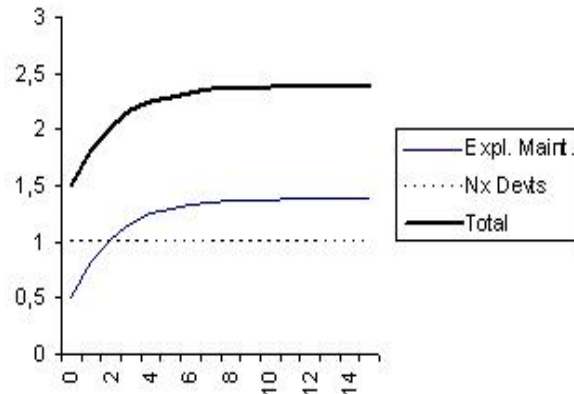
Ainsi, si le coût des nouveaux développements est constant, on obtient à terme un budget informatique constant (figure 15.7). Cette politique a un effet stabilisateur sur le coût de l'informatique mais les entreprises qui n'auront pas correctement anticipé le coût de la maintenance peuvent être surprises par son montant. Les volumes, eux, croissent exponentiellement au rythme de la baisse des coûts unitaires.

Part des nouveaux développements dans le budget constante

Supposons constante la part des dépenses consacrée au développement de nouvelles applications dans le budget informatique : $c = b_t N_t / C_t$ avec $0 < c < 1$ (dans la simulation nous supposons que $c = 60\%$).

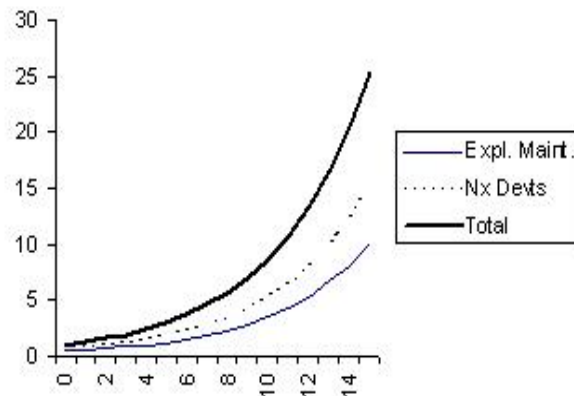
On démontre que :

$$\frac{C_{t+1} - C_t}{C_t} = \frac{\Delta C}{C} = (1 - k) \left[1 - \delta + \frac{ac}{b(1 - c)} \right] - 1$$

FIG. 15.7 – *Budget constant pour les nouveaux développements*

Le budget informatique a donc une évolution exponentielle. Sous les hypothèses retenues ici, cette exponentielle est croissante (figure 15.8). Si l'on supposait une baisse des coûts plus rapide (par exemple $k = 40\%$) elle serait décroissante.

Cette politique a donc un effet déstabilisateur sur le budget de l'informatique : il aura une tendance exponentielle, explosive ou implosive selon les coefficients économiques.

FIG. 15.8 – *Part des nouveaux développements constante*

Budget constant

Dans ce cas $C_t = c$. Cette règle est celle qui est plus fréquemment invoquée par les directions générales, qui la retiennent souvent après avoir tenté en vain de faire décroître le budget de l'informatique.

Le coût d'exploitation $a_t S_t$ tend alors vers :

$$(a_t S_t)^* = \frac{ca}{b[1 - (1 - k)(1 - \delta)] + a(1 - k)}$$

Le coût des développements $b_t N_t$ tend vers $(b_t N_t)^* = c - (a_t S_t)^*$.

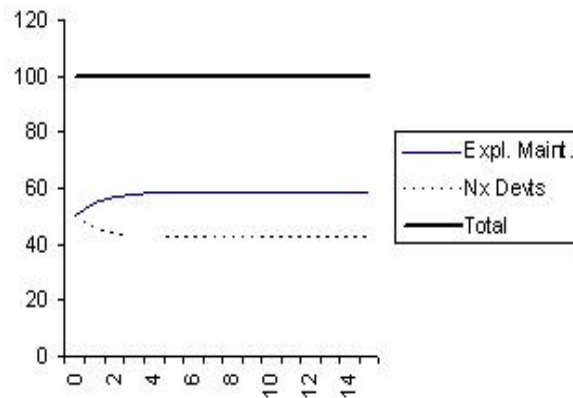


FIG. 15.9 – Budget total constant

Cette règle stabilise donc non seulement le montant du budget, mais aussi la part qu'y occupent et l'exploitation, et les nouveaux développements (figure 15.9).

Il est utile d'examiner une variante où l'on suppose la baisse des coûts k et l'obsolescence δ toutes deux nulles. On trouve alors l'évolution que représente la figure 15.10.

La règle « budget constant » contraint donc, si l'obsolescence est faible et la baisse des prix négligeable, à *annuler les dépenses consacrées aux développements* : la totalité du budget informatique étant à terme consacrée à la maintenance et à l'exploitation, l'innovation est devenue impossible. Cette règle, qui peut paraître *a priori* raisonnable et qui est en fait souvent appliquée, peut conduire à fixer les nouveaux développements à un niveau sans rapport avec les besoins de l'entreprise.

Coût de la maîtrise d'ouvrage

Nous n'avons jusqu'à présent considéré que les dépenses proprement informatiques car c'est sur elles que l'attention se focalise en général. Le coût d'un système d'information comprend aussi les dépenses de la maîtrise d'ouvrage. Elles se classent selon les rubriques suivantes :

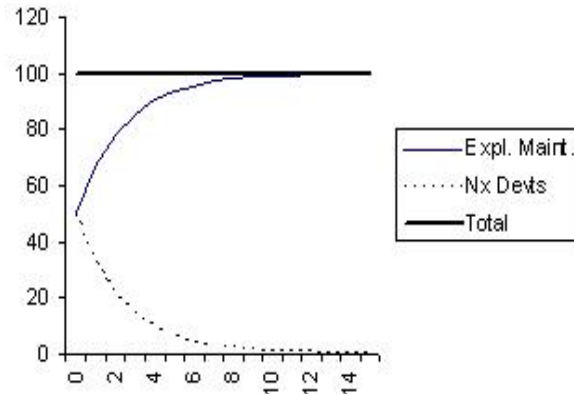


FIG. 15.10 – Budget total constant, variante

- dépenses de suivi du système d'information existant, que l'on peut supposer proportionnelles à son volume ; leur montant pendant l'année t est donc une proportion $\alpha a_t S_t$ du coût de maintenance et d'exploitation, et nous retiendrons ici $\alpha = 5\%$;

- dépenses de mise en place : formation des utilisateurs, déploiement, conduite du changement ; on peut affecter ces dépenses à l'année qui suit l'investissement informatique ; leur montant lors de l'année t est donc $\beta b_{t-1} N_{t-1}$, et nous retiendrons $\beta = 20\%$;

- dépenses de suivi des réalisations : suivi du projet et recettes. On peut affecter ces dépenses à l'année de réalisation : leur montant pendant l'année t est $\gamma b_t N_t$, et nous retiendrons $\gamma = 10\%$;

- dépenses de conception : expression des besoins, étude économique et sélection des projets, spécifications générales, validation des spécifications détaillées. On peut affecter ces dépenses à l'année qui précède l'investissement informatique ; leur montant pendant l'année t est donc $\varepsilon b_{t+1} N_{t+1}$, où nous retiendrons $\varepsilon = 25\%$.

Le coût A_t de la maîtrise d'ouvrage est donc une fraction des dépenses consacrées à l'exploitation et à la maintenance du stock des applications existantes, augmenté d'une moyenne mobile pondérée des dépenses consacrées à la production de nouvelles applications :

$$A_t = \alpha a_t S_t + \beta b_{t-1} N_{t-1} + \gamma b_t N_t + \varepsilon b_{t+1} N_{t+1}$$

Lorsque le régime de croisière est atteint, les dépenses de la maîtrise d'ouvrage représentent dans les divers cas que nous avons exploré de l'ordre de 20 % du coût total du système d'information - ou, ce qui revient au même, de l'ordre de 25 % du coût de l'informatique.

À l'issue de cet exercice, nous voyons que l'application de règles de gestion simples à l'informatique peut entraîner des effets imprévus :

- constance de la valeur des nouveaux développements : le budget informatique tend vers une limite C^* . Cette politique a donc un effet stabilisateur, toute la question étant de savoir si la dépense C^* est optimale ou non.

- part constante des nouveaux développements dans le budget : cette politique est déstabilisatrice car le budget total a une évolution exponentielle, explosive ou implosive selon les valeurs des coefficients.

- budget total constant : les parts consacrées aux nouveaux développement et à la maintenance tendent toutes deux à se stabiliser. Cette politique est par nature stabilisatrice en termes budgétaires mais elle peut avoir des effets extrêmes selon les coefficients (il peut arriver que le budget disponible pour les nouveaux développements soit fortement comprimé par les coûts de la maintenance et de l'exploitation).

Les simulations ont exploré les conséquences de règles de gestion appliquées de façon rigide. Dans la pratique, une entreprise modifiera ses règles pour éviter la catastrophe : derrière la succession des règles se profilera implicitement la recherche d'un optimum.

15.3.5 À la recherche du taux d'informatisation optimal

Les règles simples que nous avons évoquées ne sont qu'une approximation grossière de ce que pourrait être la dynamique optimale du système d'information. Nous allons tenter de donner un contenu à cette dernière expression en conservant les mêmes notations que ci-dessus en ce qui concerne le coût du système d'information²⁶ et en utilisant les méthodes de l'optimisation dynamique (Intriligator [89]). Cela nous obligera à faire un peu de mathématiques mais nous obtiendrons un résultat intéressant.

Nous supposons :

- que la fonction de production de l'entreprise est une Cobb-Douglas à rendement constant :

$$Y = F(S, L) = S^\mu L^\nu, \text{ avec } 0 < \mu < 1, \quad 0 < \nu < 1, \quad \mu + \nu = 1,$$

où Y est la mesure du *volume* produit par l'entreprise, S la taille du système d'information, L l'effectif salarié ;

- que le prix unitaire p auquel le produit Y est vendu sur le marché est égal à 1 ;

- que le salaire annuel est w ;

- que l'effectif L des salariés croît au rythme annuel θ ;

- que l'entreprise cherche à maximiser la fonction objectif :

$$W = \int_{t_0}^{t_1} (Y_t - wL_t - C_t)^\xi e^{-rt} dt, \text{ avec } \xi \leq 1,$$

26. Nous supposons que le coût de la maîtrise d'ouvrage est pris en compte.

où r est le taux d'actualisation.

$\Pi_t = Y_t - wL_t - C_t$ est la valeur du profit annuel. Si $\xi = 1$, W est la valeur actualisée du profit. Si $\xi < 1$, l'entreprise accorde d'autant plus d'importance au profit que celui-ci est plus bas, ce qui correspond à un comportement répandu.

Nous pourrions retenir d'autres spécifications mais la Cobb-Douglas est la fonction de production que les économistes utilisent le plus souvent²⁷.

Nous supposons résolus les problèmes d'efficacité que pose la mise en œuvre des facteurs : Y sera la production maximale possible avec les quantités S et L (on suppose donc que les méthodes utilisées pour construire et exploiter le système d'information sont les meilleures, qu'il s'agisse de la qualité et du contrôle des logiciels ou de la maintenance etc.). Il ne reste alors plus qu'à déterminer la stratégie d'informatisation la plus efficace compte tenu du coût des facteurs, et cette stratégie se résume par le choix de la série chronologique S_t .

* *

Associions à toutes les quantités (S , N etc.) une quantité par personne représentée par une lettre minuscule ($s = S/L$, $n = N/L$ etc.).

La fonction de production s'écrit alors :

$$y = \frac{F(S, L)}{L} = f(s) = s^\mu$$

La variable d'état sera le « taux d'informatisation » s .

Supposons le prix des unités d'œuvre constant ($k = 0$). Dans ce cas le budget informatique par tête est :

$$c_t = as_t + bn_t$$

La variation du taux d'informatisation entre l'année t et l'année $t+1$ est égale au volume des nouveaux développements par personne, moins la perte due à l'obsolescence, moins l'effet de la croissance du nombre des personnes :

$$\frac{ds}{dt} = n_t - (\delta + \theta)s_t$$

Le profit par personne lors de l'année t est :

$$\pi_t = y_t - w - c_t$$

en éliminant n_t entre ces relations on obtient l'équation du mouvement :

$$\frac{ds}{dt} = \frac{f(s) - w - \pi}{b} - \lambda s, \text{ avec } \lambda = \frac{a}{b} + \delta + \theta$$

27. On démontre que si les entreprises d'un secteur ont une fonction de production à facteurs complémentaires, le jeu de la concurrence et de la libre entrée fera que le secteur, considéré dans son ensemble, a pour fonction de production une Cobb-Douglas à rendement constant (Volle [213] p. 49).

L'état initial est caractérisé par le niveau s_0 du taux d'informatisation à l'instant t_0 , l'état final par le niveau s_1 à l'instant t_1 . La variable de contrôle est le profit par personne π ; le problème est de trouver sur l'intervalle (t_0, t_1) la trajectoire π_t qui maximise la fonction objectif tout en satisfaisant à la fois l'équation du mouvement, les valeurs initiale et finale de s et la contrainte $0 \leq \pi \leq f(s) - as - w$.

Le « théorème du péage²⁸ » indique que le taux d'informatisation suivra une trajectoire qui, à partir de s_0 , rejoindra rapidement un niveau s^* , dit « taux d'informatisation optimal », dont elle ne s'écartera que peu de temps avant de rejoindre le niveau s_1 . Les simulations ci-dessous illustrent ce phénomène. Si les valeurs initiales et finales s_0 et s_1 de s sont égales à s^* , la « trajectoire optimale » est celle qui maintient le taux d'informatisation au niveau s^* .

Le calcul²⁹ fournit enfin le taux d'informatisation optimal :

$$s^* = \left[\frac{\mu}{a + b[r + \delta + \theta(1 - \xi)]} \right]^{\frac{1}{1-\mu}}$$

Le long de la trajectoire optimale, le budget informatique par personne est

$$c^* = s^*(a + b(\delta + \theta))$$

et le profit par personne est $\pi^* = s^{*\mu} - c^* - w$. Les flux annuels s'en déduisent : la taille du système d'information est $S_t^* = L_t s^*$, la production est $Y_t^* = L_t s^{*\mu}$, le volume des développements est $N_t^* = L_t(\delta + \theta)s^*$, le budget informatique est

$$C_t^* = L_t s^*(a + b(\delta + \theta)),$$

le profit est $\Pi_t^* = Y_t^* - C_t^* - wL_t$.

Le couple (s^*, π^*) définit le *sentier de croissance optimal*. Le long de ce sentier, la production Y_t , le profit Π_t et la taille du système d'information S_t croissent au même taux θ que le nombre L_t des salariés.

* *

Ainsi, et sous les hypothèses que nous avons retenues sur la fonction de production et le coût du système d'information, le taux d'informatisation optimal existe et il est unique. L'entreprise qui le respecte maximise l'efficacité de son système d'information. Si elle s'en écarte elle est soit sous-informatisée, soit sur-informatisée.

L'expression de ce taux résulte des hypothèses. La relation ci-dessus ne saurait donc être utilisée sans précautions par une entreprise réelle, ne

28. *Turnpike theorem*, Intriligator [89] p. 435.

29. Le détail de ce calcul se trouve à l'adresse www.volle.com/travaux/annexesioptimal.htm

serait-ce que parce qu'il est difficile de spécifier sa fonction de production. *Il faut considérer cette relation comme un résultat purement théorique*, mais il fournit des enseignements utiles :

- comme il existe un taux d'informatisation optimal, les politiques qui consisteraient soit à annuler l'informatisation, soit à l'accroître indéfiniment, ne sont ni l'une ni l'autre efficaces ;

- le taux d'informatisation optimal est fonction croissante de μ , élasticité de la production à la taille du système d'information $\mu = (S/Y)(\partial Y/\partial S)$;

- le taux d'informatisation optimal est fonction décroissante du prix a de la maintenance, du prix b des développements, du taux d'obsolescence δ , et si $\xi < 1$ du taux de croissance des effectifs θ ; si $\xi = 1$, c'est-à-dire si la fonction objectif de l'entreprise est le profit actualisé, le taux d'informatisation optimal ne dépend pas du taux de croissance des effectifs ;

- le taux d'informatisation optimal ne dépend pas du taux de salaire w (mais l'entreprise ne peut exister que si le niveau de w lui permet de dégager un profit positif) ;

- comme en général $r + \delta + \theta(1 - \xi) < 1$, le taux d'informatisation optimal est plus sensible au coût de la maintenance qu'au coût du développement. *Le coût de la maintenance est donc celui auquel il convient d'être le plus attentif* : cette indication est généralement admise par l'intuition, mais on en tire rarement les conséquences tant l'attention est accaparée par les projets.

Les résultats ci-dessus sont relatifs au cas où les coûts unitaires du système d'information ne diminuent pas ($k = 0$). Lorsque $k > 0$ il n'est pas possible de résoudre le problème de façon analytique mais nous pourrions le traiter par une simulation numérique.

Simulations numériques

Prenons les valeurs suivantes des coefficients : $\mu = 0,5$; $\delta = 0,1$; $r = 0,1$; $\xi = 0,8$; $a = 0,5$; $b = 1$; $\theta = 0$; $w = 0,2$.

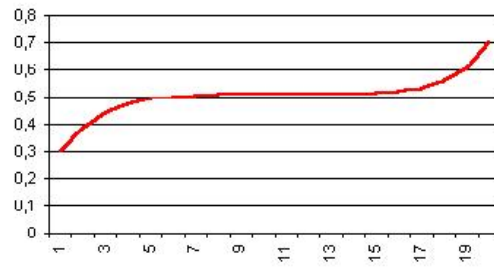
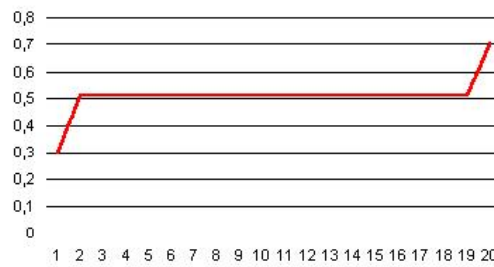
Si $k = 0$, on trouve $s^* = 0,5102$. Supposons que les valeurs initiales et finales sont $s_0 = 0,3$ et $s_1 = 0,7$. Une simulation sur vingt ans fournit la série s_t^* optimale. Son allure fait apparaître le phénomène de péage (figure 15.11).

Si l'on suppose que $\xi = 1$ (la fonction objectif est le profit actualisé), le phénomène de péage est brutal : le taux d'informatisation rejoint la valeur s^* dès l'année 1 et ne s'en écarte que la dernière année (figure 15.12).

Supposons que le coût unitaire du système d'information diminue et posons $k = 10\%$. Le taux d'informatisation (en volume) suit alors une évolution exponentielle : il croît à proportion de la baisse du coût unitaire.

On constate qu'à l'optimum la part du coût du système d'information dans la valeur de la production est constante. Ce résultat évoque un autre phénomène de péage, la variable d'état étant non plus le taux d'informatisation s_t mais la part C_t/Y_t du budget informatique dans la valeur ajoutée.

La simulation numérique apporte donc, outre la visualisation du phénomène de péage, les résultats suivants :

FIG. 15.11 – *Evolution du taux d'informatisation*FIG. 15.12 – *Evolution du taux d'informatisation si la fonction objectif est le profit actualisé*

- le phénomène de péage est d'autant plus brutal que ξ est plus élevé (c'est-à-dire que l'entreprise raisonne davantage en terme de profit actualisé) ;

- lorsque le coût du système d'information diminue ($k > 0$) la variable d'état qu'il convient de contrôler est non le taux d'informatisation s_t , mais la part du budget informatique dans la valeur ajoutée ; le taux d'informatisation optimal (en volume) croît alors de façon exponentielle.

Chapitre 16

Pathologie de l'entreprise

« Ce n'est pas la faute du miroir si tu as la gueule de travers¹ »
(Nicolai Gogol, *Le Revizor*, 1836).

16.1 Sociologie de l'entreprise

Toute entreprise est un être vivant et concret. Comme tel, elle peut être vue sous divers aspects dont elle assure la synthèse. Richard Feynman disait qu'on trouve le monde dans un verre de vin : on trouve un monde dans chaque entreprise.

Le plus souvent on considère l'entreprise sous l'angle économique (fonction et facteurs de production, fonction de coût, marché, investissement, dimensionnement des équipements, différenciation des produits, créances et dettes) ou l'angle juridique (propriété du capital, contrats de travail, contrats commerciaux) ; on peut la considérer aussi sous l'angle sociologique pour observer les échelles de valeurs, rapports d'influence, sphères de légitimité, conflits de pouvoir etc.

L'approche sociologique est parfois jugée suspecte. D'une part elle fait l'objet de l'essentiel des conversations de couloir et de cantine dans l'entreprise : c'est donc du « Café du Commerce », ce n'est pas sérieux. D'autre part certains sociologues la désavouent : pour eux, la société forme un tout et on ne doit donc pas séparer la sociologie de l'entreprise de celle qui concerne la société tout entière².

Ceux qui travaillent sur le système d'information d'une entreprise doivent cependant se colleter avec sa sociologie, avec des phénomènes qui influencent

1. На зекрало неча пеняць, коли рожа крива.

2. Autrement dit, ils estiment que l'on ne peut pas analyser la sociologie d'une entreprise particulière sans se référer explicitement aux acquis de la science sociologique concernant la société entière. Étudiez posément, longuement, la façon dont votre entreprise fonctionne, puis tentez de communiquer le résultat de vos réflexions à un sociologue : vous n'aurez pas parlé depuis quinze secondes qu'il vous aura interrompu pour rectifier votre propos. C'est là un travers commun à beaucoup de spécialistes.

comportements et décisions et que l'on ne peut réduire ni à l'économie, ni au juridique : si le système d'information a, comme nous le pensons, quelque chose à voir avec le langage que l'on parle dans l'entreprise, avec les priorités et les valeurs qu'impliquent son organisation et sa stratégie, il sera en relation avec sa sociologie.

Toute entreprise possède une sociologie particulière et nous ne pourrons pas rendre compte de cette diversité ; nous allons nous limiter à quelques « vues » qui, comme autant de petits modèles, fournissent un cadre conceptuel qu'il faut enrichir et préciser dans chaque cas.

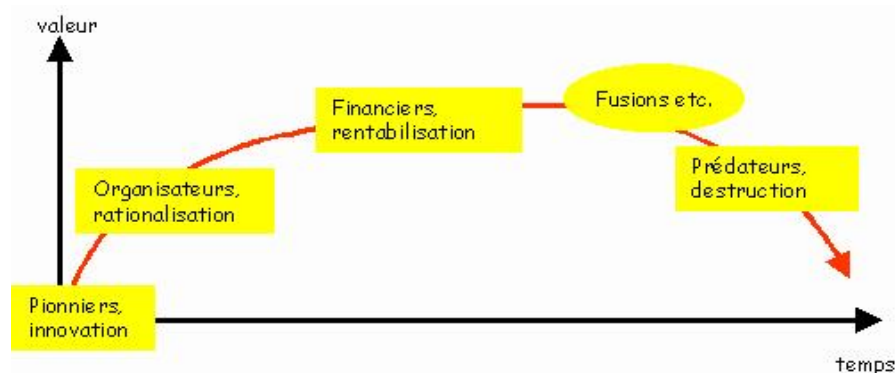


FIG. 16.1 – Cycle de vie de l'entreprise

Comme tout être vivant, l'entreprise a une histoire : elle naît, elle croît, elle meurt. L'âge d'une personne est un paramètre important de son caractère ; de même il est utile de savoir où l'entreprise en est dans sa trajectoire pour comprendre les rôles et ambitions des personnes qui la composent, l'articulation des structures de décision etc. Il est utile aussi de *réaliser* que l'entreprise mourra un jour : on la croit trop souvent éternelle. Traçons donc la trajectoire à grands traits.

Créée par des *pionniers* qui pèsent risques et opportunités, l'entreprise est à l'origine modeste et aventureuse. Son système d'information est d'autant plus souple et évolutif qu'il reste léger, l'entreprise étant de petite taille. Pour l'essentiel, elle se contente de la bureautique et d'un Intranet.

Après le succès, la formule des pionniers est érigée en recette par des *organiseurs* qui en feront une paisible mais efficace routine (ainsi Ray Kroc a systématisé et généralisé la formule de restauration rapide mise au point par Maurice et Richard McDonald). Les pionniers s'ennuient et partent. Le système d'information s'est alourdi et structuré : des « applications » sont apparues, qu'il faut maintenir et faire évoluer. Une DSI est née.

Puis l'entreprise prospère et dégage une trésorerie qu'il faut gérer. Les *financiers* arrivent. Ils transformeront l'entreprise en vache à lait. Désormais ses dirigeants ne cherchent plus à « changer le monde » mais à « faire du business ». Ses cadres savent que pour réussir il faut se conformer aux dogmes maison et ne pas faire de zèle.

Les financiers recherchent des montages (fusions, absorptions etc.) dont ils attendent une « création de valeur » (c'est-à-dire, dans leur langage, une montée du cours de l'action) mais qui souvent compromettent l'économie de l'entreprise : les réussites sont rares car les conditions physiques du montage sont rarement réunies.

Le système d'information fournit les données nécessaires au contrôle de gestion. La DSI est devenue l'un des pôles de pouvoir de l'entreprise. Les partenariats échouent souvent en raison de la difficulté à interconnecter les systèmes d'information.

À force de croître, l'entreprise devient enfin énorme. Des *réseaux* politiques, syndicaux, corporatistes l'enserrent pour y pomper richesse et pouvoir. Ils réagissent devant la nouveauté, la réflexion, comme des reptiles d'autant plus dangereux que leur cerveau minuscule abandonne le travail à la moelle épinière. Gare au naïf dont l'initiative touche un point du crocodile : il sera fauché par un mouvement réflexe.

L'entreprise, prisonnière des réseaux, devient rigide comme une personne atteinte de myosite ossifiante progressive³. Elle mourra s'il se produit des chocs extérieurs car elle ne pourra pas leur répondre. Son système d'information est, lui aussi, devenu rigide : il serait coûteux de le modifier et toute modification susciterait des conflits.

* *

Toute entreprise traverse, durant son histoire, des situations dont ce scénario illustre la diversité. La direction générale est le théâtre de conflits dont l'enjeu est, à travers l'entrelacement des intérêts particuliers, la *personnalité* de l'entreprise. Sur le terrain et au jour le jour, cette personnalité semble stable comme la surface d'un lac qui cache courants et tourbillons, les échos des conflits internes à la direction s'estompant avec la distance.

Le pire n'est cependant jamais certain car des marges de manœuvre existent. Si l'entreprise est un être vivant, elle n'est pas dotée comme le sont les plantes et les animaux d'une horloge biologique qui fixe l'échéance de sa décrépitude : elle peut se rajeunir et rebondir. Cependant l'innovation sera d'autant plus difficile que l'entreprise est plus grande et que les réseaux d'influence l'enserrent plus étroitement. On tente parfois de faire passer l'innovation par le système d'information, mais si l'organisation de l'entreprise s'y oppose cela ne pourra pas aboutir.

16.1.1 Style de la direction

Quelles sont les valeurs auxquelles l'entreprise obéit ? comment, en fonction de quels critères, se distribuent le pouvoir de décision, le pouvoir d'influence, le droit à la parole ? Toujours dans les grandes lignes, nous distinguerons trois styles : le système des caciques, le système rationalisé, le système organique.

3. La « maladie de l'homme de pierre » est caractérisée par l'ossification du tissu musculaire.

*Système des caciques*⁴

L'entreprise est dirigée par des « anciens » dont chacun a, durant sa carrière, construit son réseau de relations et négocié sa zone d'influence. Le directeur général est un arbitre qui veille à l'équilibre des pouvoirs en donnant raison (et budget) tantôt à l'un, tantôt à l'autre. Il divise pour régner.

L'énergie de l'entreprise se consume en négociations internes. Les qualités demandées au personnel sont discipline, dévouement, fidélité, égalité d'humeur. Les compétences, acquises avant d'entrer dans l'entreprise, y progressent peu car elles ne constituent pas un critère d'avancement.

Cette entreprise ne peut survivre que si son marché est stable ou si elle est protégée. C'était le cas des monopoles publics avant que la concurrence n'arrivât, c'est le cas de certaines entreprises protégées par un monopole local et de certaines administrations.

Le système d'information est construit autour d'applications pérennes, gérées par une DSI elle-même dirigée par un cacique. L'innovation est rare car les idées nouvelles sont jugées inopportunes.

Système « rationalisé »

L'entreprise est divisée en centres de résultat dotés chacun d'objectifs et de comptes permettant d'évaluer son efficacité. Pour construire la comptabilité analytique, il a fallu définir des conventions (prix d'ordre, imputation des charges communes etc.) âprement négociées ; une fois ces choix faits la négociation portera surtout sur la décision d'investir, que le calcul éclaire sans ambiguïté sinon sans incertitude.

Les responsabilités sont décentralisées. L'organigramme, qui définit les entités et désigne leurs responsables, est la pièce maîtresse de l'organisation. Il doit être assez stable pour que l'on puisse confronter engagements et résultats.

Ce système convient à des entreprises produisant en série des produits standards sur des marchés à évolution lente. Il facilite la gestion des infrastructures, l'organisation d'une force de travail spécialisée, la préparation des plans d'investissement. Les compétences demandées sont des savoir-faire correspondant chacun à des tâches bien définies. L'entreprise dispense les formations nécessaires ; les qualifications étant standardisées, les individus sont interchangeables.

Le système d'information sera volontiers structuré autour d'un ERP (voir page 289). Il fournit des éléments d'aide à la décision et s'organise autour du référentiel de l'organisation, colonne vertébrale de la comptabilité analytique. Cependant il est difficile d'organiser des projets transverses : il est pratiquement impossible de trouver le centre de résultat qui accepterait de porter une dépense nécessaire pour l'entreprise, mais qui risquerait d'avoir un effet négatif sur ses propres comptes.

Le passage du système des caciques au système rationalisé se fait, sous la pression de la concurrence, pour diminuer les coûts et restaurer la marge. Il

4. « Cacique : chef des indigènes » (Littré).

implique l'élimination des caciques, la mise en place de centres de résultats et de procédures de planification. Il transforme les critères de gestion et les points de repère du personnel.

Système « organique »

Pour l'entreprise « organique », le mot clé est *processus*. Un processus comporte une succession de décisions qui doivent être prises par les agents opérationnels. Le contrôle hiérarchique joue *a posteriori* et répond aux éventuels dysfonctionnements en modifiant l'organisation du processus. La responsabilité est décentralisée vers les exécutants. La hiérarchie est courte, ce qui facilite en principe le contact entre base et sommet.

Les agents opérationnels se qualifient en travaillant. Les qualités qui leur sont demandées sont l'adaptabilité (pouvoir activer des processus divers), le bon sens (prendre la décision juste face à un cas particulier), l'esprit de responsabilité (assumer les décisions sans angoisse). Le système d'information met en œuvre l'« aide à la tâche » (travail assisté par ordinateur).

L'entreprise rationalisée résistera d'autant plus à la mise en place du système organique qu'elle est mieux organisée. Pour une entité jugée sur ses résultats, tout processus qui traverse sa frontière doit en effet être muni de compteurs. Mais comment évaluer une expertise? Et si plusieurs entités coopèrent à un même processus, comment en partager la responsabilité et le résultat?

Il avait fallu casser le système des caciques pour passer au système rationalisé ; il faut casser le système rationalisé pour passer au système organique. Chacun de ces passages suppose sacrifices et destructions.

16.1.2 Un reflet des valeurs

L'entreprise s'exprime dans sa communication interne, son rapport annuel, sa publicité. Mais ces discours décrivent ce qu'elle voudrait être ou paraître, non ce qu'elle est. Elle voudrait être aimée par de fidèles clients, admirée pour la qualité de ses produits, prestigieuse par son professionnalisme. Il faut, pour voir ce qu'elle est vraiment, pénétrer l'arrière-cuisine où elle s'exprime naïvement et sans même le savoir. Cet endroit, c'est le système d'information. En examinant celui-ci on découvrira des priorités différentes de celles affichées ou souhaitées.

La vérité de l'entreprise, c'est bien souvent d'abord le *nombrilisme*. Quoique l'on prétende, ce n'est pas le client qui est « au cœur de l'entreprise » mais l'organisation interne. La priorité des *managers* ne réside pas dans le marché mais dans leurs plates-bandes mutuelles.

Cela se voit dans les identifiants. Identifie-t-on le client, dont on veut suivre le comportement et étudier les besoins, ou bien l'équipement dont l'entreprise se sert pour produire le service? le marketing dit qu'il faut identifier le client ; la pratique, c'est que l'opérateur télécoms identifie la ligne téléphonique, le banquier le numéro de compte, l'hôpital le lit. Votre banquier ne sait pas additionner les crédits et débits de vos divers comptes, votre opérateur télécoms vous envoie autant de factures que vous avez de lignes etc. L'entreprise est dévorée par l'organisation interne, à laquelle elle

accorde toute son attention ; le client passe après⁵. L'étonnante mauvaise qualité des centres d'appel en est une conséquence (voir page 568).

La relation avec la clientèle n'est pas la seule victime de la priorité accordée à l'organisation. Dans une entreprise industrielle, on trouvera une nomenclature de produits différente dans chaque usine, dans chaque filiale : cela rendra difficile la définition de produits intégrés comme la mise en place de l'« e-business ». Le découpage géographique du marché sera dicté par l'équilibre des pouvoirs entre barons ou par des règles conventionnelles de gestion (« pas plus de cinq établissements pour une direction régionale ») aux dépens de la compréhension du marché.

Il n'est pas surprenant alors qu'il soit si difficile de réussir les partenariats où l'entreprise associerait, en un même assemblage, son offre à celle d'autres entreprises. La réussite d'un partenariat suppose que les systèmes d'information échangent les données nécessaires au processus de production et, surtout, au partage du profit. Mais il sera pratiquement impossible de faire communiquer des systèmes d'information dont les fondations reflètent non l'économie du marché mais l'organisation particulière à chaque entreprise.

16.1.3 Pôles de légitimité

« Les friponneries, les scélératesses, comment les fortunes se font et se ruinent, l'art et les détours pour arriver à ses fins, tromper, gouverner, museler les rois, se faire des partis et des créatures, écarter le mérite, l'esprit, la capacité, la vertu, en un mot les manèges des cours. » (Saint-Simon, *Mémoires*, Gallimard, Bibliothèque de la Pléiade 1987, vol. VII p. 521)

La décision tranche les débats, arbitre les conflits, indique la stratégie. C'est la fonction du « décideur ». On ne peut évoquer la décision dans l'entreprise sans parler de *légitimité* : seul est habilité à décider celui qui en a légitimement la fonction. L'expert a le droit, et même le devoir, de donner un avis ; mais ce n'est pas à lui qu'il revient de décider.

On distingue plusieurs sortes d'expertise : l'expertise du terrain, apportée par la première ligne ; l'expertise des « concepteurs métier », apportée par des ingénieurs attentifs à l'état de l'art professionnel ; l'expertise technique, apportée par d'autres ingénieurs qui suivent l'état de l'art technique⁶. La qualité de la procédure de décision réside dans la distinction des rôles des experts et du décideur et dans le soin apporté à leur articulation (voir page 421).

5. Les choses évoluent : certaines banques ont redéfini leur système d'information de façon à pouvoir rassembler toutes les données relatives à un même client (et, accessoirement, lui permettre de les consulter sur la Toile). Les opérateurs télécoms se débattent avec l'héritage de leur système de facturation pour pouvoir « reconstruire le client ». Mais ces modifications sont lourdes, lentes, et rencontrent encore de fortes résistances.

6. Dans le cas du système d'information, l'expertise du terrain vient des utilisateurs ; les concepteurs métier sont les maîtres d'ouvrage opérationnels ; l'expertise technique est fournie par les informaticiens.

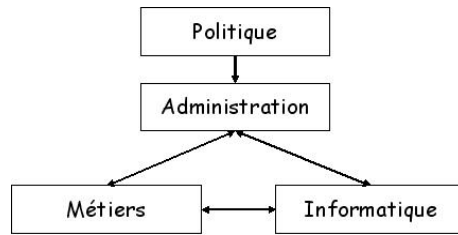


FIG. 16.2 – Les quatre pôles de légitimité

La légitimité se distribue autour de quatre pôles (figure 16.2) : le président-directeur général possède la légitimité *politique* ; l'administration est légitime sur les questions d'*organisation* ainsi que pour l'*évaluation* des projets ; les métiers sont les maîtres de l'expertise *professionnelle* ; l'informatique possède l'expertise *technique*.

* *

Le président de l'entreprise incarne, comme le fait la Couronne en Grande-Bretagne, la légitimité suprême. Il se charge des relations avec les actionnaires, les banquiers, l'administration ; dans les grandes entreprises, il est en rapport avec le gouvernement. Il représente l'entreprise devant les médias.

Le dirigeant français n'est généralement pas un ingénieur mais plutôt un politique : depuis les années 1970, le pouvoir dans les grandes entreprises a été pris par des inspecteurs des finances ou par des personnes qui singent les inspecteurs des finances (il existe des exceptions, mais encore une fois nous schématisons).

Ce dirigeant estimerait incompatible avec son statut social de mettre les mains sur un clavier d'ordinateur : ce serait pour lui aussi exotique que de prendre le métro. S'il a une boîte aux lettres électronique, c'est son assistante qui trie et imprime les messages. Même quand il dit être conscient de l'importance « stratégique » du système d'information, il n'en a qu'une notion vague et il est mal à l'aise pour arbitrer les décisions le concernant.

Il arrive qu'il soit volontariste (« je veux que nous soyons présents sur l'e-commerce dans six mois », disait un président-directeur général de mes amis au début de l'année 2000), mais devant les difficultés ce volontarisme se dégradera en velléités. Les contraintes pratiques, techniques du système d'information lui paraissent mesquines, comparées à ses grandioses attentes stratégiques : si vous lui parlez du poste de travail des utilisateurs, du référentiel de l'organisation, des processus à équiper, il vous trouvera bien terre-à-terre.

À l'orée de l'économie de l'« immatériel » et des « services », certains dirigeants en viennent à penser que la technique n'a aucune importance, l'essentiel étant affaire d'image de marque⁷. Ils surestiment alors l'import-

7. « La technique, moi, je n'en ai rien à foutre » a déclaré Michel Bon lors d'une conférence à Dauphine. Venant du président d'une entreprise qui exploite un gigantesque

tance des médias, de la communication, au détriment de la physique de l'entreprise.

* *

L'administration de l'entreprise, c'est l'ensemble constitué par la direction financière, le contrôle de gestion, la direction des ressources humaines etc. Ce sont des activités de contrôle et d'organisation, de support et de normalisation. En relation directe avec la direction de l'entreprise, elles relayent sa stratégie. Les responsables de l'administration ne sont généralement pas des ingénieurs mais des juristes, des économistes ou des sociologues. Pour eux aussi, le système d'information est une chose abstraite, même si se conformant à la mode ils le disent « stratégique ».

L'administration est jalouse du pouvoir que l'informatique tire de son gros budget. Elle pratiquera donc volontiers une politique de compression des coûts : quelles que soient les priorités et contraintes, le budget de l'informatique ne devra pas dépasser un montant fixé à l'avance, l'« enveloppe ». Celle-ci sera parfois libellée non en euros, mais en hommes-mois de développement et le budget proprement dit sera alors entouré d'un flou qui facilite les dérives. Sa discussion se réduira à une séance lors de laquelle le directeur général, pour manifester sa mauvaise humeur, barre quelques projets dans une longue liste.

L'administration préfère que l'organisation soit simple : elle s'oppose donc à la constitution de pôles de compétence en maîtrise d'ouvrage, enchaînant des phrases fameuses : « tout ça, c'est de l'informatique », « la maîtrise d'ouvrage doit être faite par l'informatique », « l'administration des données est un travail intellectuel, *donc* superflu ».

Il est difficile pour ces organisateurs d'instaurer dans l'entreprise à propos du système d'information, qu'ils comprennent mal, une dialectique entre deux pôles professionnels qui seraient chargés l'un des aspects fonctionnels, l'autre des aspects techniques. Une telle dialectique existait naguère entre la direction de la production et la direction commerciale, elle constituait la respiration de l'entreprise ; mais elle était subie, et non voulue, alors que la constitution d'une maîtrise d'ouvrage professionnelle suppose que l'on *veuille* créer un tel dipole.

Dans son utilisation des systèmes d'information, l'administration est souvent maladroite. Elle produit pour la direction les tableaux de bord et aides à la décision mais il est rare qu'ils soient de bonne qualité car elle n'a pas les compétences nécessaires en statistique⁸ (voir page 379).

* *

Par « métiers » nous entendons les directions qui produisent et qui vendent : marketing, commerce, production, maintenance, exploitation, dis-

automate, cette affirmation faisait froid dans le dos.

8. Le savoir statistique est utilisé par les banques pour le calcul financier, par les entreprises industrielles pour la maîtrise de la qualité, mais les « datawarehouses », « datamining » et autres systèmes d'aide à la décision en sont au tout début dans la plupart des autres entreprises.

tribution etc. C'est dans les métiers que réside la physique de l'entreprise : ils sont le lieu où se construisent ses produits et sa relation avec le marché. Ils détiennent le savoir technique et économique mais à la façon dont des consommateurs détiennent le savoir sur leurs besoins : s'ils en sont les porteurs, cela ne veut pas dire qu'ils sachent les exprimer. Étant accaparés par l'activité quotidienne il leur est d'ailleurs souvent difficile de définir leurs priorités, de les communiquer entre eux et à la direction de l'entreprise.

La conception du système d'information suppose que les métiers traversent dans le bon ordre trois couches différentes. Il faut d'abord que les utilisateurs « réalisent », c'est-à-dire qu'ils perçoivent que la solution existe *réellement*. Il faut ensuite qu'ils comprennent comment cela marche. Ce n'est qu'après ces deux étapes qu'ils peuvent assimiler le savoir-faire et apprendre à s'en servir. Supposez que vous installiez une messagerie dans une entreprise qui en était auparavant dépourvue. La première difficulté pour l'utilisateur sera de « réaliser » en quoi la messagerie consiste : il découvre avec étonnement la possibilité d'ouvrir les messages en cliquant dessus, d'envoyer une réponse, l'existence du carnet d'adresse et des pièces jointes etc.

S'il est impossible d'utiliser un outil dont on n'a pas « réalisé » l'existence, il n'est par contre pas absolument indispensable que l'utilisateur comprenne comment il fonctionne : d'excellents conducteurs ignorent comment fonctionne leur automobile ; par contre cette compréhension est nécessaire aux maîtres d'ouvrage, aux concepteurs qui doivent définir l'outil, le paramétrer, l'articuler à d'autres au sein du système d'information.

Enfin, une fois ces étapes franchies (l'étape « comprendre » pouvant être légère pour certaines personnes), on peut passer à l'acquisition du savoir-faire, des réflexes qui permettront d'utiliser l'outil sans même y penser et comme s'il était tout naturel.

La mise en place des workflows se heurte à cette difficulté intellectuelle : bien que la conception et la réalisation d'un workflow ne soient pas plus délicates que d'autres démarches, il est difficile pour une personne qui n'en a jamais vu de « réaliser » en quoi cela consiste. Je garde un souvenir cuisant de certains exposés où je me démenais comme un pasteur en plein sermon mais ne recevais en retour que des regards vides, incrédules ou hostiles.

Un métier aura du mal à modéliser son activité, à définir les priorités entre ses besoins, parce qu'il lui est difficile de « réaliser » par avance les services que le système d'information lui apportera. C'est cela qui rend si délicate l'utilisation d'UML par les maîtrises d'ouvrage ainsi que la validation des modèles par le maître d'ouvrage stratégique.

Enfin le système d'information apporte des changements à l'organisation des métiers et cela suscite parfois l'appréhension. La transparence, que tout le monde souhaite en principe, n'est pas la bienvenue pour ceux dont elle compromet le pouvoir ; la suppression des niveaux hiérarchiques intermédiaires est souhaitée par tous, sauf par ceux dont le poste sera supprimé.

* *

L'innovation réussie, c'est la clé du succès pour l'entreprise (voir page 172) : elle lui permet de diminuer le coût de production, d'offrir des produits

nouveaux, de prendre de l'avance sur les concurrents, de dégager un surprofit qui réjouira ses actionnaires et facilitera le financement des investissements.

Certaines grandes entreprises possèdent à cette fin un centre de recherche, ou bien elles ont des relations assidues avec des chercheurs. Mais les chercheurs, les innovateurs, forment un petit monde à part qui a ses propres valeurs, signes de reconnaissance, effets de modes et priorités. La sociologie de l'entreprise a certes une influence sur la sociologie des chercheurs mais cette influence reste lointaine : le chercheur dialoguera plus volontiers avec un homologue étranger qu'avec les métiers de l'entreprise, ce qui les fait d'ailleurs enrager.

L'innovation se produira dans les domaines que les chercheurs ont fouillés avec le plus d'intensité. Le choix de ces domaines est donc décisif. Sans doute le flair des chercheurs les oriente vers les plus prometteurs, mais il n'obéit pas qu'à des critères scientifiques : il existe des préjugés dans ce milieu, des conformismes ; un centre de recherche est le théâtre de conflits de pouvoir parfois aussi violents que ceux de la direction générale. Il en résulte qu'une part de l'effort de recherche reste stérile, l'innovation étant le produit fatal d'une activité qui ne la visait pas en priorité.

Quoiqu'il en soit, des innovations se produisent. Elles ne sont pas immédiatement les bienvenues car elles provoquent dans les métiers des chocs parfois épouvantables. Elles modifient le champ du possible. Il faudra d'abord que l'on « réalise » cette modification, que l'on en prenne la mesure, que l'on assimile sa réalité. Il faudra mettre en production le produit nouveau, dont l'industrialisation comporte des difficultés et des coûts parfois dix fois supérieurs à ceux de la recherche. Puis il faudra modifier les processus de production, les circuits de vente etc.

L'innovation tombe sur les métiers comme un météorite. L'entreprise refusera longtemps de la mettre en œuvre : pendant des années, Xerox a refusé l'imprimante à laser, les opérateurs télécoms ont longtemps refusé le téléphone mobile etc.

* *

L'informatique est au sein de l'entreprise un pôle de pouvoir d'autant plus fort que le fonctionnement des métiers dépend davantage d'elle et que son budget est plus important. Une entreprise de service de quelques dizaines de milliers de personnes dépense pour l'informatique plusieurs centaines de millions d'euros par an : c'est, avec les frais de personnel et l'immobilier, un des plus gros postes de dépense. Le directeur de l'informatique est un potentat.

Pourtant l'informatique souffre. Elle a été soumise à une évolution rapide : remplacement de la mécanographie par les ordinateurs dans les années 1960, mise en réseau, temps partagé, terminaux et mini-ordinateurs dans les années 1970, micro-ordinateurs dans les années 1980, développement des réseaux locaux, des applications bureautiques et de l'Internet dans les années 1990. Les plates-formes (machines, systèmes d'exploitation) ont été bousculées ainsi que les langages de programmation (Fortran et surtout Cobol font place à C++, Java et Perl).

À chacune de ces étapes des personnes ont été mises sur la touche. Des activités que sottement l'on croit ancillaires, comme la programmation, sont devenues de plus en plus compliquées sans acquérir pour autant un meilleur statut social. Par ailleurs il est devenu impossible à une DSI de maîtriser l'ensemble des savoirs nécessaires à son activité : pour choisir dans la diversité des solutions, maîtriser les langages et interfaces, évaluer la pérennité de fournisseurs dont elle dépendra, elle doit faire appel à des SSII. Elle est alors confrontée à des séductions parfois trompeuses : il est rare qu'un fournisseur tienne intégralement ses promesses une fois le contrat signé. Les montants en jeu étant importants, il en résulte des tentations fortes.

La sociologie de l'informatique est donc en crise et sans doute en transition vers une nouvelle informatique dont on ne voit pas clairement ce qu'elle sera. Cependant les informaticiens d'aujourd'hui ont gardé de leur histoire certains traits qui les caractérisent. Par formation, ils restent attachés aux grands systèmes centraux et s'ils participent au déploiement de la bureautique, c'est sans plaisir. L'informatique de communication (messagerie, documentation électronique) ne correspond pas à leur culture qui s'attache plutôt au traitement des données structurées. Si le système d'information est fécondé par la rencontre entre l'informatique de communication (qui véhicule le langage connoté, symbolique, des êtres humains) et l'informatique de traitement des données structurées (qui repose sur un langage conceptuel qui facilite la modélisation), cette rencontre ne s'opère pas aisément dans la sociologie de l'informatique.

L'expérience montre combien il est difficile, pour quelqu'un qui a été formé à la rigueur des langages conceptuels, de s'accommoder du flou des connotations alors que celui-ci est si efficace dans la communication entre personnes. L'informaticien de formation parle souvent une langue de bois ; il lui est difficile de comprendre, de sentir les situations et les personnes : c'est là une vraie maladie professionnelle, même si bien sûr certains informaticiens en sont exempts.

* *

Personne ne reconnaît exactement son propre comportement dans une étude sociologique ; de même, personne ne reconnaîtra exactement son entreprise dans les esquisses que nous venons de présenter. Leur intérêt n'est pas tant de décrire des situations concrètes que d'illustrer des perspectives, des points de vue que chacun pourra utiliser pour interpréter la situation de sa propre entreprise : à quel point de sa trajectoire se trouve-t-elle ? Quel est le régime de son organisation (caciques, rationnel, organique ?) Quelle est la nature des relations entre les pôles de légitimité ?

Il est vrai que nous n'avons pas procédé ici avec la rigueur scientifique que réclame la sociologie : nous n'avons pas fait d'enquête ni établi de statistiques. Nous nous sommes contentés d'interpréter une expérience, de l'enrichir par la confrontation avec d'autres expériences. Nous ne nous sommes donc pas beaucoup éloignés du Café du Commerce mais peut-être celui-ci est-il trop décrié. Il faut bien en effet, avant que ne se mettent en place les enquêtes et méthodes rigoureuses, que des naïfs posent des jalons conceptuels vers un point de vue inédit ou du moins peu fréquenté.

16.2 Crise de l'entreprise

La cause première d'une crise de l'entreprise, *c'est un processus de décision perturbé* : lorsque les missions sont mal définies, les personnes nommées à contre-emploi, les processus de production négligés, l'action est désorientée et la nature se venge par des incidents qui, non résolus, s'accumulent.

Incidents classiques en cas de crise

- procédures longues et erratiques ;
- délais et budgets incontrôlables et dépassés ;
- négociations sans fin ;
- remise en question des décisions ;
- désaveu des mandataires ;
- ajournement des rendez-vous ;
- absentéisme en réunion ;
- réunions sans ordre du jour, durée limite ni compte rendu ;
- pannes fréquentes sans responsable identifié ;
- messagerie et courrier infidèles ;
- documentation non tenue à jour ;
- substitution d'une « certification qualité » à la vigilance ;
- informatique désordonnée (redondance des référentiels, ressaisies manuelles, ergonomie pénible).

Dans n'importe quelle entreprise l'un ou l'autre de ces incidents peut se produire à l'occasion. Ce qui révèle une crise, c'est le fait qu'ils se produisent de façon systématique, automatique, et que même s'ils suscitent protestations et mauvaise humeur celui qui tente d'y mettre un terme est considéré comme un Don Quichotte.

À titre d'exemple, voici une panne survenue dans le *hub* d'un transporteur aérien. Pendant une matinée, les postes de travail n'ont pas pu accéder aux applications centrales et l'enregistrement a été bloqué : allongement des files d'attentes, retards au décollage, embouteillage de l'aéroport, protestations des passagers exaspérés. Lorsque la direction générale a demandé des explications à la DSI, celle-ci a répondu que la CPU s'était arrêtée pendant cinq minutes.

Restait à expliquer l'écart entre une panne de cinq minutes et un arrêt de service de plusieurs heures : c'est que l'arrêt de la CPU avait provoqué, en cascade, l'arrêt de plusieurs serveurs ; un des serveurs se trouvait dans un local fermé à clé ; la personne qui savait où se trouvait la clé était en congé : cela rappelait les aventures de Tintin⁹...

9. Dans *L'île noire*, des pompiers cherchent longuement la clé du local où est rangée

La DSI se souciant de la seule l'application centrale, personne n'était responsable de la continuité du service rendu à l'utilisateur final ; pour que cette continuité fût assurée il aurait fallu redéfinir les missions et territoires de certaines directions et une telle redéfinition n'était pas à l'ordre du jour. Les choses ont donc repris leur cours sans qu'aucune mesure ne soit prise, le risque de panne restant suspendu au dessus des agents opérationnels comme l'épée de Damoclès.

* *

Lorsqu'il y a crise de l'entreprise, chacun cherche non à résoudre les problèmes - tâche devenue impossible - mais à en faire porter par d'autres la responsabilité. Les dirigeants sont comme quelqu'un qui, perdu dans le désert, marcherait dans la direction opposée à celle du salut : même s'ils sentent où se trouve la solution, sa simplicité leur inspire une sorte de crainte et ils changent son signe comme s'ils permutaient le Sud et le Nord d'une boussole. Le meilleur, ils le rejettent ; le pire, ils le choisissent.

La crise se manifeste par un symptôme qui ne trompe pas : le langage que l'on parle dans l'entreprise se dégrade (voir page 556). Si le vocabulaire est pollué par des synonymes et homonymes, si les données que fournit le système d'information sont ambiguës ou, ce qui revient au même, s'il faut des redressements préalables pour les utiliser, les réunions seront encombrées de discussions stériles. L'incertitude sur les faits entraîne l'imprécision de l'action et la légèreté des décisions.

La « langue de bois » empêchant l'évaluation pondérée des problèmes, il faut un bouc émissaire. On le cherche, on le trouve. Son exécution (mise au placard, dépression, départ) se prépare dans son dos avec fébrilité et jubilation. Elle procure une détente momentanée mais ne résoud rien : il faudra donc le remplacer par une nouvelle victime expiatoire. Les énergies de l'entreprise en crise se consomment dans la destruction des personnes.

* *

Peut-on soigner le mal en corrigeant le langage ? Non, car il n'est qu'un symptôme. Si l'on restaure le processus de décision, par contre, le langage suivra.

Cependant le langage des dirigeants conditionne ce processus : c'est à eux en effet qu'il revient de « donner du sens » à l'entreprise. Il s'agit non de lui apporter un « supplément d'âme », expression un peu méprisante, mais de consolider l'architecture de l'entreprise : processus de production et de gestion, outils de connaissance et d'interprétation, élaboration de la stratégie. Or pour une entreprise, comme pour un bâtiment, ce qui a été construit s'impose par son évidence et sa pérennité apparente ; ce qui se conçoit, puis se construit, relève de l'imagination et de la volonté. Les personnes qui manquent d'imagination ne voient que l'existant (elles appellent cela du *pragmatisme*) et ignorent le possible.

la pompe à incendie.

Manifestations de la crise du langage

Quand les mots « sérieux », « rigueur », « professionnalisme », « méthodologie » reviennent souvent, c'est signe que ces qualités font défaut : celui qui est vraiment sérieux ne perd pas de temps à les prononcer.

Ceux qui ont peur d'être en position de faiblesse si on les comprend se protègent par du jargon (abondance d'acronymes, d'anglicismes, de néologismes, de faux amis, de noms propres).

Certains remplaceront le mot propre par un terme abstrait ou décalé (« méthodologie » pour méthode, « problématique » pour problème, « technologie » pour technique, « générique » pour général, « spécifique » pour particulier ou pour local, « commanditaire » pour donneur d'ordres, « ordonnancement » pour mise en ordre etc.) : les dégâts sont ici plus graves que ceux que cause le jargon car c'est le vocabulaire courant lui-même qui est dégradé.

Le recours habituel au superlatif (« très grave », « très important », « très sérieux ») voire au double superlatif (« c'est très très important ») aplatit le langage : le manque de contraste interdit la perception des priorités.

Certaines expressions s'autodétruisent (« principes concrets », « schéma exhaustif », « synthèse détaillée » etc.) : dans ces bombes sémantiques l'explosif est la contradiction entre le substantif et l'adjectif (nécessairement un principe est abstrait, un schéma sélectif etc.). Une de ces bombes est l'expression « pilotage stratégique » : le pilotage, qui implique action et réaction à court terme, est par nature tactique et il faudrait dire « pilotage opérationnel » et « orientation stratégique ».

La dégradation des concepts accompagne celle du langage : on dit « organisation » mais on dessine un organigramme ; « processus » sans définir ni produit ni acteurs ; « qualité » (ou mieux « méthodologie de démarche qualité ») sans préciser les critères d'évaluation ^a.

On confond données comptables (biaisées notamment par le principe de prudence) et indicateurs économiques ; gestion (suivi de l'opérationnel) et expertise ; observation (constat des faits) et explication (utilisation d'un modèle) ; donnée statistique (sur une population) et donnée individuelle.

^a Quand quelqu'un parle de qualité, il faut lui demander de donner sa définition de la qualité et d'indiquer les critères dont il se sert pour l'évaluer : on découvrira alors souvent qu'il ne sait pas de quoi il parle.

L'histoire, en montrant les origines de l'existant, permet de percevoir la dynamique de sa conception ; cependant les conformistes seront imperméables à ses enseignements. Mieux vaut donc, en utilisant une démarche analogue à celle de l'école de Palo Alto [160], progresser par petits changements pour faire percevoir puis accepter de nouvelles perspectives.

Il ne s'agit alors que de débloquer les conditions pratiques les plus élémentaires du processus de décision : tableaux de bord sélectifs et clairs pour les dirigeants ; comités équilibrés, où l'expertise s'exprime et où la légitimité décide (voir page 421) ; qualité sémantique du système d'information ; équilibre des responsabilités et des pouvoirs entre maîtrise d'ouvrage et maîtrise d'œuvre etc.

Nous ne détaillerons pas davantage ces conditions : il s'agissait seulement ici de montrer comment l'examen du langage contribue au diagnostic de crise. Si l'entreprise fonctionne comme un organisme sain, son langage sera sans enflure ; on ne parlera plus de « sérieux », de « méthodologie », de « qualité » ; les agents opérationnels diront : « la boîte marche bien », « c'est simple », « on sait ce qu'on a à faire », « on est bien outillés », « on est bien dirigés », « c'est efficace », « ça tourne », « c'est organisé » etc. Cette clarté, cette simplicité sont le meilleur actif de l'entreprise. Il se construit progressivement, puis s'entretient, par un processus de décision fin et vigilant.

16.3 Obstacles au changement

On entend souvent une phrase qui révèle l'importance des obstacles à surmonter : « pour mettre en place un système d'information, il faut que le directeur général s'implique personnellement ». Il n'en est pas de même dans les autres domaines de l'entreprise : si les décisions d'investissement, d'organisation, sont soumises à l'approbation du stratège, personne ne dit que dans ces divers domaines les choses ne pourront avancer que si celui-ci s'implique.

On entend dire aussi que l'amélioration du système d'information rencontre des « obstacles culturels ». Cela signifie que comme il touche aux valeurs, aux habitudes, à la *personnalité* de l'entreprise, sa mise en œuvre suscite des questions aussi confuses que profondes et provoque des résistances instinctives. Pourquoi cela ?

* *

Le système d'information alimente les tableaux de bord de l'entreprise et lui permet d'élaborer une expertise. On peut représenter cette construction par un modèle en couches. À la base se trouve l'enregistrement des faits qui peut s'opérer de façon automatique à partir des processus opérationnels (communications des clients pour un opérateur télécoms, tickets de caisse pour un magasin, coupons de vol pour un transporteur aérien etc.). À partir de cet enregistrement élémentaire l'entreprise peut produire des indicateurs sur la demande, les coûts, le partage du marché etc. Cela suppose qu'elle sache estimer les données manquantes, analyser les séries chronologiques,

corriger le mouvement saisonnier, extraire la tendance, segmenter la clientèle, ce qui nécessite une compétence en production statistique du côté des métiers, en utilisation de la statistique du côté des dirigeants. Or beaucoup de dirigeants confondent économie et comptabilité et n'ont aucune compétence en statistique¹⁰.

La recherche opérationnelle utilise les indicateurs pour modéliser les comportements de la demande, des concurrents, des fournisseurs, explorer des hypothèses et éclairer la stratégie. Le responsable qui aura longuement travaillé les données, testé les hypothèses, exploré les scénarios, disposera ainsi d'une représentation claire de son domaine d'action. Le système d'information peut par ailleurs fournir aux agents opérationnels, dans une fenêtre qui s'ouvre sur l'écran, des indications permettant de traiter chaque client en fonction du segment auquel il appartient ; il fournit aux superviseurs une aide qui soulage leur charge mentale lorsque les problèmes d'exploitation s'accumulent etc.

Ces diverses expertises s'édifient à partir des modèles que fournit la recherche opérationnelle. Or il arrive que l'entreprise demande au système d'information de fournir cette expertise (dans les têtes des dirigeants, dans les outils de la première ligne, dans l'aide aux superviseurs) sans pour autant la soutenir par une statistique ni par une recherche opérationnelle, comme si l'expertise pouvait découler directement de l'enregistrement des faits (figure 16.3).

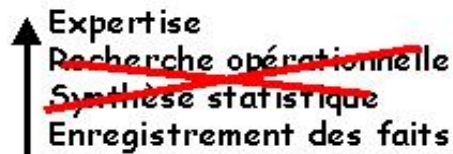


FIG. 16.3 – L'expertise à partir des faits bruts

C'est comme si un opérateur télécoms voulait faire communiquer ses clients par télépathie, comme si un transporteur aérien voulait faire voyager ses passagers sur un tapis volant. Ces solutions ingénieuses fonctionnent en effet, mais seulement dans *Les mille et une nuits*.

Pour expliquer cette situation étrange mais fréquente, nous proposons de représenter l'entreprise selon un modèle en trois couches : en bas se trouve la couche *physique, économique* des « métiers », avec la fonction de production, la formation de la demande, la formation des prix, les relations de concurrence, les partenariats etc. En haut se trouve la couche *politique* où se nouent les relations avec les puissances extérieures et où se négocie le

10. Ils préféreront souvent un indicateur fallacieux comme « rapport entre le dernier mois et le mois correspondant de l'année précédente » qu'il est impossible d'interpréter mais qui, semble-t-il, leur « parle » davantage qu'une série corrigée des variations saisonnières. Ils aiment bien aussi les « comparaisons entre prévision et réalisation », en nommant « prévision » le chiffre établi lors de la préparation du budget annuel.

crédit accordé à l'entreprise : actionnaires, banquiers (et en outre, dans le cas des grandes entreprises publiques, ministère de tutelle, gouvernement, parlement, « Bruxelles », direction du Trésor etc.). Entre les deux se trouve une couche de *représentation* où se découpent les concepts qui permettent à l'entreprise de se décrire, de se connaître, de communiquer : cette couche n'est autre désormais que le système d'information, qui fournit le cadre conceptuel, l'outil d'observation et de synthèse dont l'entreprise a besoin pour être pensable, partageable, communicable, mémorisable (figure 16.4).

Ensemble, les couches *Économie* et *Représentation* forment le « Système économique » de l'entreprise ; les couches *Politique* et *Représentation* forment le « Système politico-financier » ; la couche *Représentation* est commune aux deux systèmes, qu'elle articule et fait communiquer.



FIG. 16.4 – Les trois couches de l'entreprise

Dès lors existe le risque d'une coupure : le système politico-financier peut s'émanciper du système économique auquel il demandera seulement d'alimenter le jeu politique. Alors la couche représentation se coupe en deux (figure 16.5). La partie qui colle à la politique, et qui définit à la fois les sphères de légitimité et les images par lesquelles elles se font reconnaître, devient *médiatique* : l'économie de l'entreprise y est représentée par un double symbolique qui, tout comme l'image d'un homme politique, d'une vedette, a une vie indépendante de celle de l'être qui en est support ou prétexte.

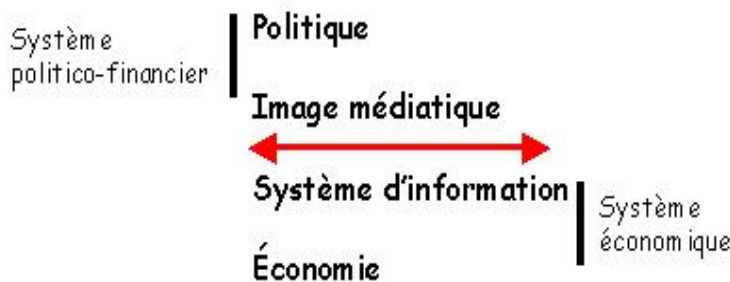


FIG. 16.5 – Coupure entre la physique et la politique

Le partage du pouvoir entre dirigeants à l'intérieur de l'entreprise obéissant alors à la logique médiatique, les zones de légitimité se défendent par des procédés symboliques : qui figure ou non sur la liste de diffusion de telle

note? qui participe ou non à telle réunion? M. X prend-il ou non M. Y au téléphone? dans quel délai M. X accorde-t-il un entretien à M. Y si celui-ci le lui demande? Voilà les questions importantes!

Le fait qu'une entreprise n'accorde pas d'importance à l'observation des faits, à leur synthèse, à l'analyse des tendances, à la confrontation des modèles explicatifs révèle que pour ses dirigeants la légitimité se limite à leur image auprès des actionnaires, des banquiers ou du gouvernement. La coupure entre le milieu des dirigeants et le fonctionnement physique de l'entreprise est manifeste lorsque le système d'information ne parvient pas à communiquer avec les dirigeants. La pierre de touche est la qualité des tableaux de bord (voir page 379).

Il arrive alors que les couches physique et politique coexistent en menant leur vie chacune de son côté, les frictions ne se produisant que lorsqu'une décision économiquement nécessaire bute sur un refus à motivation politique, ou si une intention politique rencontre un obstacle physique. Si l'entreprise sait éviter ce type de situation, si l'environnement ne lui est pas trop hostile, elle peut prospérer ainsi.

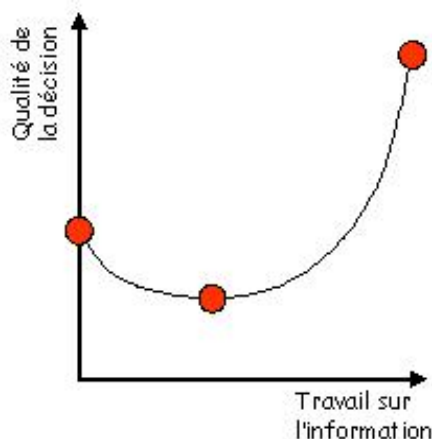


FIG. 16.6 – *Système d'information et qualité de la décision*

Les réflexes des dirigeants attentifs à la physique de l'entreprise sont affûtés par l'examen anticipé des scénarios possibles, par la connaissance des ordres de grandeur et des réactions du marché. La qualité de leurs décisions est élevée; nous les avons représentés par le point en haut à droite sur la figure 16.6¹¹.

Ceux qui ne se soucient pas de l'information et ne regardent pas les tableaux de bord décident « au pif »¹². Il n'en résulte pas nécessairement

11. Un bon exemple de ce type de dirigeant est fourni par Bob Crandall, d'American Airlines (Petzinger [159]).

12. L'exemple vient du sommet: « Le mécanisme de prise de décision me paraissait, depuis longtemps, constituer un des points faibles de notre manière de gouverner » (Valéry

que la qualité de leurs décisions soit médiocre : l'intuition, nourrie par des contacts informels avec des collaborateurs et par des visites sur le terrain, peut leur permettre d'éviter les plus graves erreurs. Nous les représentons par le point à gauche de la figure 16.6. Leurs décisions ne peuvent cependant avoir ni l'énergie, ni la précision, ni la cohérence que l'on trouve chez ceux qui sont attentifs aux faits.

Supposons qu'un tel dirigeant, conscient de ses lacunes, se mette à compulsurer des statistiques, à regarder des tableaux et des courbes. *Cet effort méritoire aura d'abord un effet négatif* : la fraîcheur, le bon sens qui soutenaient son intuition sont momentanément détruits sans qu'il soit pour autant devenu un expert. L'arbre lui cache la forêt, tel détail vu dans les données le préoccupe à l'excès. La qualité de ses décisions a baissé et ses collaborateurs regrettent le temps où il travaillait moins mais où il était plus raisonnable. Il se trouve au point bas de la figure 16.6.

Celui qui commence à regarder les données passe en effet par une phase pénible durant laquelle ses décisions seront moins bonnes, son intuition moins fidèle. L'effort finira par payer s'il est poursuivi mais la transition aura été rude. C'est pourquoi la position située à gauche de la figure 16.6 est un point d'équilibre, même s'il n'est pas optimal, car le rendement marginal du travail y est négatif. Cette position sera presque toujours adoptée par les dirigeants qui, étant nommés par un pouvoir politique, sont novices en matière de stratégie.

* *

Dans une entreprise où la direction est essentiellement politique, les métiers eux-mêmes sont incités à s'écarter de la rigueur professionnelle et à obéir aux règles classiques du conformisme : « pas de vagues », « pas vu, pas pris », « après moi le déluge ».

Ceux qui évitent les vérités désagréables trouveront les paragraphes ci-dessous de mauvais goût, d'autres y reconnaîtront certaines de leurs expériences.

« *Pas de vagues* »

Le premier devoir d'un responsable est de couvrir les fautes de ses subordonnés ; il n'y a jamais de sanctions - sauf envers ceux qui faisant « des vagues », « du zèle », ont fait apparaître des problèmes que l'on aurait préféré ignorer.

L'acceptation institutionnelle du mot « compétence » s'écarte de l'usage courant. En français courant, la personne compétente est celle qui possède le savoir nécessaire à sa fonction. En français administratif, la personne compétente pour traiter une question est celle dont cette question relève selon l'organigramme. Lorsque la compétence administrative entre en conflit avec la compétence du savoir, c'est à la première que l'on donnera raison, sinon ce serait l'anarchie.

Une innovation risque toujours d'entraîner des réorganisations et un changement des conditions d'utilisation de la force de travail. Pour com-

Giscard d'Estaing, *Le pouvoir et la vie*, Compagnie 12 1988).

battre l'innovation, on fera un épouvantail des « problèmes sociaux » qu'elle risquerait de susciter. Lors de cette manœuvre, il est opportun de se rengorger (le « goitre du dirigeant » donne à la voix un son grave) et de prendre un air *très* préoccupé.

« *Pas vu, pas pris* »

Le pouvoir ne procure vraiment de plaisir que s'il est arbitraire. Faire appliquer une décision rationnelle, ce n'est pas vraiment du pouvoir puisque ceux qui doivent l'appliquer peuvent y adhérer en se fondant sur leur raison. Les contraindre à appliquer une décision absurde, par contre, procure une pure jouissance. Il serait naïf d'ignorer le penchant de l'être humain pour de telles voluptés - qui, comme l'adultère, restent sans conséquences tant qu'elles sont indécélables.

La rétention d'information, le retard des signatures, transforment les collègues en suppliants et constituent une monnaie d'échange : c'est ainsi que l'on devient important.

Il ne faut jamais se sentir tenu par un engagement. Prendre un engagement ne coûte rien et permet de se débarrasser d'une trop forte pression. L'essentiel est que la promesse soit oubliée (ou qu'il soit de mauvais goût de la rappeler) lorsque l'engagement arrivera à échéance.

Si l'entreprise contraint à faire des *reportings*, il faut en retarder la fourniture en alléguant les difficultés de la collecte d'information et les urgences opérationnelles, et il convient d'entourer les évaluations d'un tel flou qu'elles échapperont à toute discussion.

« *Après moi, le déluge* »

La légèreté des informaticiens qui ont continué à coder les années sur deux caractères alors que l'an 2000 approchait illustre à elle seule cette rubrique. La désinvolture avec laquelle les entreprises mettent à la porte des « anciens » qui emportent avec eux la compétence des métiers, la lenteur dans l'embauche des jeunes qui apporteraient des compétences conformes à l'état de l'art en sont un autre symptôme.

L'insouciance se trahit dans les attitudes velléitaires qui associent le discours volontariste à l'immobilisme. La violence du discours est d'ailleurs un symptôme de velléité : l'homme authentiquement volontaire n'éprouve pas le besoin de se montrer violent.

* *

La conception d'un système d'information réclame une compétence en sémantique et en physique : il faut tout à la fois définir les concepts et les modèles qui alimenteront la couche « représentation » et aussi dimensionner les ressources qu'utilisera la plate-forme informatique, ce qui suppose de savoir anticiper le comportement des utilisateurs du système d'information. Traiter cela par le calcul supposerait de manier en virtuose les probabilités des trafics, des pannes, des comportements en cas de panne etc. Dans la pratique, *l'expertise remplace ici le calcul*.

L'expert est quelqu'un qui est déjà tombé dans les pièges et qui s'en est sorti, à chaud, sous les *lazzi* des utilisateurs. Il a appris à anticiper, par l'intuition, les accidents possibles sur un réseau. Cette intuition peut

parfois s'exprimer de façon simple : on peut ainsi prévoir, et dire, que si une entreprise met en place un nouveau système sans former les utilisateurs ceux-ci commettront des erreurs et que le *help desk* sera alors surchargé de questions élémentaires. Mais certaines de ces certitudes sont plus difficiles à communiquer et à partager.

Supposons que l'entreprise souhaite construire un système d'information sur sa clientèle. L'expert sait qu'il faut un répertoire pour identifier les clients, car cela permettra de rassembler au moindre coût l'information que l'entreprise possède sur un client quelle qu'en soit la source. Cependant la construction de ce répertoire a un coût et un délai et des dirigeants impatientes peuvent ne pas en percevoir l'utilité¹³.

Il en sera de même des hypercubes qui accélèrent l'utilisation des bases de données moyennant quelques limitations ; de l'administration des données et de la modélisation des processus, qui clarifient la sémantique d'une opération avant tout développement technique¹⁴ ; de l'équipement des utilisateurs en interfaces multimédia qui élargissent la gamme des fonctionnalités possibles ; du dimensionnement de l'infrastructure de serveurs et du réseau ; d'outils qui, comme le serveur de télécopie, économisent le temps et l'attention de l'utilisateur ; de l'unification des messageries qui permet à l'utilisateur de retrouver l'ensemble de ses messages dans une même boîte aux lettres ; de la documentation électronique et des forums ; de l'équipement des processus en workflows etc.

Sur tous ces sujets, l'expert est éclairé par une évidence simple, aussi puissante que celle qui s'impose à l'architecte lorsqu'il équilibre les masses en leur procurant des points d'appui ; mais sauf exception cette évidence ne sera pas partagée par les personnes inexpertes.

Si, dans une compagnie aérienne, quelqu'un proposait de faire voler les avions sur le dos « parce que les passagers trouveront cela amusant et que cela nous distinguera de la concurrence », il serait déconsidéré : chacun sait que les passagers n'apprécieraient pas cette acrobatie d'ailleurs impossible.

Dans le domaine du système d'information, par contre, l'expert entend dire des énormités qui équivalent à « faire voler des avions sur le dos » mais qui passent au comité de direction pour des hypothèses à considérer sérieusement : « on fera le répertoire en dernier », « il ne faut pas installer de serveur de fax parce que cela ferait croître la dépense en télécoms », « il ne faut pas formaliser le processus sous la forme d'un workflow parce que cela reviendrait à graver en dur les erreurs que le processus comporte », « il faut faire des économies immédiates dans l'opérationnel, la recherche de la cohérence relève d'une démarche intellectuelle et donc superflue », « la maîtrise d'ouvrage doit être faite par l'informatique », « l'administration des données peut attendre la fin du développement », « je ne crois pas à l'In-

13. J'ai connu ainsi une entreprise qui entendait « mettre le client au cœur de l'entreprise » mais qui ne savait pas identifier ses clients : le directeur général ne percevait pas cette incohérence.

14. Alors que nous travaillions à l'élaboration d'un modèle qui nous donnait bien du mal, un directeur nous demanda : « Mais alors, qui fait le travail ? » - pour lui, « travailler », c'était écrire des lignes de codes ; modéliser, ce n'était pas du travail.

ternet¹⁵ », « il n'y a qu'à externaliser notre centre d'appel en Roumanie », « l'informatique est devenue une *commodity* » etc.

Il est difficile de partager l'expertise sur le système d'information parce qu'il s'agit d'une spécialité nouvelle. En outre le système d'information associe à des tâches pratiques, physiques (faire une réparation, livrer un produit, transporter un paquet) une *représentation*. Il n'est pas facile pour un dirigeant de comprendre ce que l'on gagne en redoublant ainsi les tâches pratiques par leur image, structurée dans un cadre conceptuel : l'efficacité d'une représentation immatérielle est parfois perçue au coup par coup, mais rarement dans son principe et sa généralité.

Chacun peut à la rigueur comprendre que le calendrier de maintenance d'un équipement soit enregistré dans un programme informatique qui édite les documents techniques, produit les « fiches de travail » permettant de travailler dans le bon ordre (démonter une pièce, puis les pièces que ce premier démontage dégage, exécuter les travaux sur les pièces dans l'ordre inverse du démontage etc.), enregistre les opérations, met à jour le programme d'entretien etc. Mais il ne sera pas facile de comprendre que le système d'information est soumis à des contraintes qui lui sont propres, celle du dimensionnement des ressources et de la sémantique. Que les contraintes de la sémantique soient aussi strictes que celles de la physique, c'est un fait que peu de dirigeants sont prêts à reconnaître.

16.3.1 La tache aveugle

Lorsque l'expert exprime ce qui est pour lui une évidence relevant du bon sens, il observe le regard distrait de son interlocuteur, son empressement à parler d'autre chose ; on lui enjoint finalement d'être « plus concret ». Cet emploi du mot « concret » est révélateur : ce mot a une acception différente selon que l'on utilise le langage philosophique, où il a un sens technique précis, et le langage courant.

Dans le langage philosophique, « concret » s'oppose à « abstrait » comme « individuel » s'oppose à « conceptuel ». Est concret cet objet-ci, que je peux manipuler s'il est devant moi (cet ordinateur, cette tasse de café). Sont abstraits le point de vue à partir duquel je le considère et le concept sous lequel je vais le classer (forme, poids, couleur, matière etc.). Tout objet individuel (concret) réalise ainsi *de facto* la synthèse de diverses catégories conceptuelles (abstraites).

Mais dans le langage courant, « concret » est synonyme d' « habituel », abstrait est synonyme de « nouveau ». Rien de plus concret, pour un cadre qui s'inquiète de sa carrière, qu'une catégorie comme « C8 » qui n'a de sens que dans la grille de qualifications de l'entreprise. Les catégories fréquemment utilisées par le raisonnement (types d'actifs pour un financier, types d'outils pour un ouvrier, segmentation des clients etc.), qui sont des abstractions selon le langage de la philosophie, seront qualifiées de « concrètes ».

15. Cette dernière phrase ne s'entend plus guère en 2006, mais elle était courante dans la deuxième moitié des années 1990.

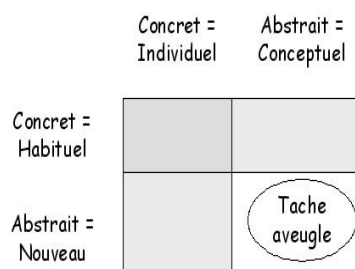


FIG. 16.7 – La tache aveugle de l'intellect

Croisons les deux acceptions des termes « concret » et « abstrait » (figure 16.7). Chacun est à l'aise dans le monde de ses objets habituels, monde en somme doublement concret. Les catégories abstraites dont on a l'habitude associent à chaque objet concret les concepts dont il relève ; elles nous semblent « concrètes » car elles délimitent nos intentions, désirs, craintes, répulsions et confèrent ainsi un sens à notre action.

Les objets individuels dont on n'a pas l'habitude, concrets au sens philosophique, se plient mal à notre perception parce que nous ne disposons pas de grille conceptuelle qui puisse en rendre compte. Nous ne saurons pas les classer par rapport à nos désirs, intentions etc., *nous ne saurons pas quoi en faire*. Si nos sens les perçoivent (pas toujours, car le cerveau fait un tri dans ce qui s'affiche sur la rétine), notre entendement ne sait comment les « penser ». Ils seront donc ignorés ou jugés « abstraits », ce qui est une façon de dire qu'ils nous mettent mal à l'aise.

Quant aux concepts dont on n'a pas l'habitude, pour nous *ils n'existent pas*. Inhabituels et imperceptibles (puisque à la différence des objets individuels ils ne se présentent pas devant les sens), ils résident dans la « tache aveugle » de l'intellect (figure 16.7). Leur évocation nous semble futile : c'est du « bavardage », du « bruit ». Nous attendons qu'elle cesse pour pouvoir revenir aux « choses réelles », à celles dont nous avons l'habitude.

Regardez ce père de famille qui parle avec son fils de dix-sept ans. L'adolescent vit avec ses copains, il pense à son habillement, à la musique qu'il aime. Si le père cherche à lui expliquer que ce qu'il apprend au lycée sera utile, plus tard, dans sa vie professionnelle, il évoque quelque chose qui est très loin de ce que l'adolescent est prêt à entendre... que le père soit clair dans ses explications n'y changera rien : lorsque l'interlocuteur est sourd, l'éloquence est inutile.

Jeanne Favret-Saada a décrit de façon exemplaire cet aveuglement devant l'inhabituel ([55] p. 233) : « Les notes que je pris en 1971 d'après la bande magnétique que j'avais enregistrée au cours de cet entretien portent alors cette mention étonnante, significative de la surdité qui m'affecta si souvent au cours de mon travail : "Suit une histoire inaudible" [...]. Il me paraît invraisemblable aujourd'hui que seul ce passage ait été inaudible : quand, plus tard, j'y entendis le ronronnement de la machine à laver des Babin, cela ne m'empêcha pas de comprendre leurs paroles. Au pire, Joséphine m'avait

alors parlé avec un débit précipité [...]. L'hypothèse la plus probable est donc que je ne voulais pas entendre le récit de cet épisode capital - sur lequel je ne posai d'ailleurs aucune question - parce que de le prendre en considération m'aurait conduit à réviser la version que je m'étais alors constituée de l'histoire des Babin ».

Le système d'information se trouve dans la tache aveugle des dirigeants parce que la formation au système d'information ne fait pas partie de leur bagage initial¹⁶, de la formation qui, leur ayant permis de « réussir », leur semble suffisante. Pour que l'entreprise assimile son système d'information et sache en faire un instrument de l'action, il faut que ses dirigeants soient à l'aise pour créer, réviser et détruire les concepts et catégories qui fondent leurs représentations - ce qui implique qu'ils soient devenus dans une certaine mesure des *penseurs* ou plus exactement qu'ils aient appris à maîtriser certains *procédés de pensée* (voir page 229). L'importance que prend le système d'information dans la vie des entreprises aidera cette évolution mais celle-ci ne sera ni facile, ni rapide.

16.3.2 Le compromis managérial

On peut distinguer parmi les entreprises celles qui sont « tirées par le marché » (les mots clés sont alors « positionnement » et « marketing ») et celles qui sont « conditionnées par l'organisation » (les mots clés sont alors « management » et « organigramme »).

Le choix entre ces deux styles dépend de la phase du cycle de vie de l'entreprise. Une entreprise jeune sera naturellement tirée par le marché. Une entreprise mûre, à la structure stable, sera par contre souvent conditionnée par son organisation (voir page 167). L'organisation, une fois stabilisée, est en effet aussi difficile à transformer que les murs d'un immeuble : pour la changer, il faudrait « tout casser » et cela coûterait cher. Les dirigeants se sentent alors contraints au « compromis managérial », au « pas de vagues » qui résume parfois toute leur sagesse.

Regardons ce qui se passe dans une entreprise conditionnée par son organisation. L'organigramme relie, du haut en bas de la pyramide, chaque responsable à un râteau de quelques collaborateurs immédiats. Une fois l'organisation définie, les « domaines de compétence » sont répartis entre ces collaborateurs. Chacun d'entre eux sera dans son domaine comme un seigneur féodal dont le château fort, campé sur une motte symbolique, entretient une guerre de coups de main contre les territoires voisins.

La première tâche du dirigeant est alors de s'assurer qu'aucun de ses collaborateurs n'empiète sur le domaine d'un autre ni ne l'inquiète par l'extension de ses responsabilités. Il est comme un roi qui veille à l'équilibre des pouvoirs et symboles entre de grands seigneurs, que l'on nomme d'ailleurs très justement les « barons ». Il doit s'accommoder des défauts des personnes et contenir leurs conflits : si l'organisation a gravé dans le marbre

16. Cela changera, mais notre économie, notre société peuvent-elles s'offrir le luxe d'attendre la relève de cette génération-là ?

un découpage inadéquat, il le laissera perdurer car le corriger susciterait des vagues. Seul capable de faire la synthèse des diverses dimensions de son propre domaine, il bénéficie d'ailleurs de la position de force qu'apporte la politique « diviser pour régner ».

Exemple de découpage inadéquat

Le directeur financier de l'entreprise a normalement, entre autres fonctions, celle d'assister le directeur général dans les décisions budgétaires qui impliquent un arbitrage entre les diverses directions.

Si l'on rattache directement au directeur financier une direction dépendante comme la DSI, il ne pourra plus assister le directeur général dans son rôle d'arbitre puisqu'il sera à la fois juge et partie - et inmanquablement il fera jouer son influence pour favoriser cette direction.

L'organigramme sculpte le vocabulaire de l'entreprise, délimite les perspectives de carrière des cadres, conditionne l'imaginaire et les projets, focalise l'attention sur le tracé des frontières. En outre la comptabilité analytique incite chaque direction à faire du profit non sur le marché, mais au détriment des autres directions. Les structures « germinales » éventuellement porteuses d'une évolution, mais faibles en budget, sont écrasées : il est exclu de leur faire place dans les réunions où se prennent les décisions, où se choisissent les orientations. La R&D est méprisée ou instrumentalisée dans les conflits internes.

Le conditionnement de l'entreprise par l'organisation se révèle lorsqu'on examine son système d'information, en particulier les conventions utilisées pour construire les référentiels. C'est lui qui explique que l'on identifie non pas le client, mais le produit et qu'on laisse perdurer dans les tables de codage un désordre nuisible à l'intégrité des données, mais favorable au maintien des frontières de l'organisation. Il explique aussi pourquoi il est si souvent difficile, voire impossible, de mettre en place une administration des données.

Lorsqu'on voit l'entreprise sous cet angle une image monstrueuse s'impose : celle d'une *personne dont la tête rejoindrait le nombril de telle sorte que sa cervelle serait aspirée par son intestin*. L'organisation, moyen de l'efficacité, est en effet devenue la finalité de l'entreprise. Alors toute réflexion sur le marché, le positionnement, la stratégie se décourage et s'éteint : « dans la position du garde-à-vous, disait Lyautey, les talons se joignent et la cervelle se vide ». La réflexion stratégique sera répudiée au bénéfice d'un prétendu « pragmatisme », terme qui désigne ici non le recours méthodique à l'expérimentation mais le fait de s'accommoder de l'organisation existante : il s'agit d'ajourner indéfiniment les conflits de frontière¹⁷.

17. Le président d'une grande entreprise a dit ainsi un jour à un de mes amis « je préfère ne pas donner suite à vos recommandations concernant le système d'information parce qu'elles poseraient des problèmes de personnes ». Leur coopération s'est arrêtée là.

L'entreprise qui ne trouve pas de ressort en elle-même pour propulser son évolution ne pourra évoluer que sous la pression de circonstances extérieures (changements de la réglementation, de la concurrence, des techniques). Ainsi s'impose une deuxième image : celle d'*une personne qui avancerait à reculons, poussée par une main posée sur sa poitrine* ; sa marche, aveugle, lente, hésitante, trébuchera sur le moindre obstacle.

Le système d'information est une des premières victimes du compromis managérial. Pour construire un système d'information efficace, il faut en effet travailler sur l'ensemble du système, sa cohérence, la qualité des référentiels qui le fondent, l'adéquation à la stratégie. Mais le dirigeant qui favorise le compromis managérial tourne le dos à l'obligation de cohérence, car elle poserait des problèmes politiques ; il encourage une approche parcellaire, chaque domaine travaillant à part et recourant à des ruses tactiques pour faire financer ses projets. Dès lors certains détails auront fait l'objet d'un soin excessif alors que des points importants restent négligés. C'est comme si pour bâtir un immeuble on avait choisi de belles espagnolettes, mais construit les escaliers en attachant les unes aux autres des chaises et des échelles dont certaines débouchent sur le vide.

Certes une entreprise à l'organisation instable aurait du mal à fonctionner. Mais comme l'organisation est un moyen au service du positionnement, de la stratégie et de l'efficacité, on doit savoir la modifier si nécessaire : l'entreprise est un agent économique, et non un de ces organismes dont la priorité est de se reproduire à l'identique.

16.3.3 Le centre d'appel : un révélateur

Un centre d'appel, c'est une partie de la première ligne, lieu de contact privilégié avec les clients ; c'est aussi le « help desk » où les problèmes des utilisateurs se manifestent ; c'est enfin le central téléphonique, porche de l'entreprise sur le réseau. Dans tous les cas, l'opérateur doit comprendre rapidement ce que lui dit une personne parfois anxieuse, presque toujours malhabile, pour qui il doit trouver une solution. Cela demande expertise et agilité d'esprit et c'est une activité très formatrice. Si vous voulez connaître les problèmes d'une entreprise, passez quelques heures au « help desk » : vous y apprendrez plus qu'en plusieurs jours d'enquête, qu'en plusieurs années à la direction générale.

Beaucoup d'entreprises considèrent pourtant le centre d'appel comme une activité secondaire. Elles le confient volontiers à des sous-traitants dont certains, en ce qui concerne le confort des travailleurs, ont pu être comparés à des élevages de poulets. Les employés des centres d'appel sont souvent mal payés ; ce sont des postes que l'on fait tenir par des étudiants stagiaires et qui connaissent un fort *turn-over*. Les agents des centres d'appel remplissent pourtant une fonction de contact que tout consultant qualifiera de « stratégique » : mais il faudrait un miracle pour que des personnels recrutés de la sorte puissent rendre un service de qualité. Les directions générales éprises d'économies de bouts de chandelle veulent croire en ce miracle.

Elles ne sont pas seules coupables. Les agents des centres d'appel ne travaillent pas dans les mêmes locaux que les autres, ne mangent pas à la même cantine, n'ont pas les mêmes horaires. Alors joue le syndrome du village gaulois : les agents du « siège » ou de la direction régionale se sentent supérieurs aux « gens du centre d'appel », qu'ils ont tendance à traiter en parias. Ces réactions instinctives, animales, ne font que conforter le comportement des directions générales.

Pourtant si l'entreprise était rationnelle elle s'y prendrait autrement. Le centre d'appel serait considéré comme le creuset où l'on doit faire passer les meilleurs éléments. L'accès aux fonctions de direction - qu'il s'agisse de diriger un établissement, une région, une direction à la direction générale - serait précédé par un passage de quelques semaines au centre d'appel pour que le futur directeur puisse voir de près les problèmes qui se manifestent dans la relation avec la clientèle, ou au « help desk » pour qu'il voie les difficultés que rencontrent les agents opérationnels. On assurerait la promotion des agents des centres d'appel vers des fonctions d'encadrement afin que l'entreprise capitalise les connaissances qu'ils ont acquises « au front » et à chaud. On analyserait les statistiques et commentaires provenant des centres d'appel, qui sont autant d'indicateurs utiles pour la stratégie.

Une entreprise qui mette un futur directeur en stage pendant quelques semaines au centre d'appel ? Mais c'est du rêve ! Nos entreprises cherchent plus à conforter le statut des dirigeants, à les initier aux aspects relationnels de leur fonction, qu'à former leur compétence. Le stage ne se fera donc pas au centre d'appel, mais plutôt dans les couloirs de la direction générale ; le futur directeur n'y apprendra rien sur les clients ni sur les utilisateurs, mais il y sera initié aux méandres de l'intrigue.

Les entreprises négligent ce que pourraient leur apprendre les observations accumulées dans le centre d'appel ; elles laissent s'évaporer, par *turn-over*, le savoir des agents qui y sont passés ; elles gaspillent un excellent moyen pour former leurs cadres. Il existe des contre-exemples, et c'est heureux¹⁸, mais la situation décrite ci-dessus est la plus fréquente : le centre d'appel est l'un des aspects du système d'information où se révèle le plus clairement la crise de l'entreprise.

16.3.4 Corruption et honnêteté

Lorsque la tentation existe, la faute est une question de probabilité : certaines personnes résistent mais d'autres succomberont. Donc quand il y a tentation il y a faute, non pas mécaniquement mais statistiquement et selon un certain pourcentage.

La plupart des crimes informatiques sont des « crimes d'occasion ». Un policier spécialisé m'en a décrit le scénario : un comptable décèle une faille dans le système ; il la signale mais personne ne s'en soucie ; la faille reste ouverte, cela le fait rêver ; il finit par « piquer » 30 € pour voir ; rien

18. Je connais une entreprise dont les dirigeants passent, de temps à autre, quelques jours au centre d'appel ou dans une agence commerciale. C'est l'exception qui confirme la règle.

ne se passe; la tentation s'accroît démesurément (« après tout, puisqu'ils s'en fichent... ») et alors il pique 300 000 €, montant moyen des crimes informatiques dont beaucoup restent impunis.

Ce policier concluait avec philosophie: « sur cent personnes, dix sont maladivement honnêtes et ne commettront jamais le moindre délit; dix sont maladivement malhonnêtes et se tiennent à l'affût des occasions; les quatre-vingt restants sont des gens comme vous et moi... ».

Il faut relativiser les affaires de corruption. C'est très mal de piquer dans la caisse mais les personnes qui l'ont fait - certes elles ont eu grand tort - ne sont pas toutes des salauds irrécupérables: il faut, pour comprendre ce qui s'est passé, évaluer la tentation à laquelle elles ont été soumises.

* *

La corruption prend des formes diverses, parfois subtiles. Les commerciaux des fournisseurs sont des gens intelligents et très sympathiques. Sous prétexte de vous faire rencontrer des experts, ils vous invitent dans d'excellents restaurants ou à des compétitions sportives intéressantes. Ils vous flattent sans aucune vergogne. J'occupais naguère des fonctions impliquant un pouvoir de prescription et les commerciaux fréquentaient assidûment mon bureau. Il étaient parfois accompagnés d'experts à qui je faisais part de mes réflexions sur les systèmes d'information. Je me rappelle le cri d'admiration que mes propos ont arraché un jour à un commercial qui, certes, de sa vie ne s'était jamais soucié de classes abstraites: « C'est génial, ce que vous venez de dire! » Il savait bien, pardi, que personne n'a jamais assommé un client à coups d'encensoir.

Il est pénible, pour le DSI d'une grande entreprise, de passer de la douche tiède de ces flatteries à la douche froide que lui administrent en réunion ses collègues, patrons des directions utilisatrices, lorsqu'ils tentent de lui faire endosser tout ce qui va mal dans l'entreprise. Rien d'étonnant si les DSI sont, parfois, des personnes un peu susceptibles et irritables.

* *

Si les DSI sont injustement maltraités par leurs collègues, ils peuvent se consoler en regardant leur budget. Supposons que vous soyez le DSI d'une entreprise de 50 000 salariés. Le budget informatique tourne autour de 300 millions d'euros, l'effectif de la DSI est de l'ordre de 2 000 personnes. Le montant du budget comprend des dépenses incontournables: salaire des personnels en place, location des équipements etc. Cependant une part significative est sujette à décision. Le *turn over* des personnels étant de l'ordre de 3 % pour de simples raisons démographiques, vous embaucherez au moins 60 personnes par an. Par ailleurs chaque projet est une occasion de contrat avec un fournisseur et il faut renouveler les équipements. Sur les 300 millions, il vous reviendra ainsi d'affecter chaque année quelques dizaines de millions.

Dès lors vous êtes une puissance. Les fournisseurs vous font la cour. Les collègues vous ménagent, car vous pourriez embaucher tel ou tel fils ou neveu. Comme vous êtes capable de rendre des services, vous pouvez aussi en demander: le troc des services rendus est la clé de bien des comportements.

L'importance des projets se mesurant par la ponction qu'ils opèrent sur votre marge de manœuvre, vous concentrerez votre attention sur les très gros projets (quelques dizaines de millions d'euros) et sur les gros projets (quelques millions d'euros); vous regarderez à l'occasion certains petits projets (quelques centaines de milliers d'euros), par contre les projets qui ne coûtent que quelques dizaines de milliers d'euros ne vous effleureront pas¹⁹.

Pourtant l'utilité d'un projet n'est pas proportionnelle à son coût. Il se peut que tel outil, qui ne coûte que quelques dizaines de milliers d'euros, soit très important pour les utilisateurs: vous ne le percevrez pas. Les fournisseurs, c'est bien naturel, vous présenteront de préférence des solutions lourdes qui se situent dans la gamme des prix « sérieux ». Vous les étudierez, vous les connaîtrez, finalement vous les préférerez. C'est pourquoi, contrairement à la théorie économique qui voudrait que l'entreprise minimisât ses coûts, la DSI évitera souvent les solutions peu onéreuses.

Il arrive aussi parfois, même s'il ne faut pas s'exagérer ce phénomène, que de l'argent aille dans des poches où il n'a rien à faire. Les réseaux politiques, syndicaux, corporatistes qui enserrent la grande entreprise s'alimentent d'emplois fictifs, de faveurs en nature et de billets de banque, la seule contrainte étant la discrétion: « pas vu, pas pris ». L'arrivée de la bureautique dans l'entreprise a décentralisé le pouvoir de décision: avec la grande informatique, seule la tête pouvait être tentée; avec la bureautique la tentation est descendue jusqu'à des niveaux relativement modestes. Un fournisseur m'a dit un jour que le « tarif », dans telle grande entreprise, était de 1 % du coût du contrat. Vu les montants en jeu cela faisait une belle somme.

Le plus souvent toutefois la corruption matérielle, grossière, n'est pas nécessaire: la flatterie est moins compromettante et moins coûteuse et si elle peut suffire, ce sera autant d'économisé pour le fournisseur.

* *

Après un séjour dans le service public qui m'avait accoutumé à une stricte déontologie, j'ai dirigé des entreprises de conseil. J'ai eu d'excellents clients que je respecte mais j'ai aussi loupé des contrats que j'aurais dû avoir. Je m'en étonnais naïvement: comment le client pouvait-il refuser une proposition d'un coût modique et de la plus grande utilité pour son entreprise? Une partie de l'explication résidait dans la modicité du coût, cf. ci-dessus. Une autre dans un manque de compréhension: le client ne « réalisait » pas ce que nous disions, comme cela s'est souvent produit pour les projets de workflow. Mais je ne peux pas exclure une troisième hypothèse: dans certains cas, si j'avais su émettre le signal annonciateur d'un cadeau, les choses se seraient mieux passées.

J'ai vu parfois le client potentiel se mettre dans une colère noire au beau milieu d'une discussion calme et professionnelle. Il m'a fallu du temps pour en comprendre la raison. Mettez-vous à la place du malheureux: je

19. Dans la construction aéronautique, où l'on compte par centaines de millions, la cécité commence au million d'euros.

déroule devant lui une argumentation imparable mais, comme je n'émet pas le signal attendu, il ne pourra pas accepter mon offre et il ne pourra pas non plus dire pourquoi il la refuse. De plus en plus mal à l'aise, il explose pour clore la discussion. Cette colère soudaine, apparemment inexplicable, est un message que j'ai fini par savoir déchiffrer.

La tentation suscite aussi le désir de *paraître* honnête, forme pernicieuse de malhonnêteté. Des entreprises se font certifier ISO 9000 pour être irréprochables : cela ne garantit pas contre la faillite. Il arrive aussi que, pour éviter le soupçon de malhonnêteté, on n'ose pas faire le bon choix technique parce qu'il est proposé par un fournisseur dont les méthodes commerciales ont mauvaise réputation.

* *

Les qualités qui aident à résister à la tentation forment une constellation identifiable : sens de l'humour et des proportions, esprit clair, goût de la concision et du travail bien fait, courage. L'honnêteté professionnelle, c'est l'addition de la compétence et de ce dévouement au métier qui fait percevoir l'erreur comme une faute contre l'évidence et contre la nature. Il peut arriver qu'un chirurgien compétent sabote un patient qu'il juge sans importance sociale, mais le chirurgien dévoué à son métier ne sabotera jamais personne.

On n'en a pas fini avec l'exigence morale quand on a les mains propres. Sommes-nous assez courageux ? assez volontaires ? assez vigilants ? nous laissons-nous aller à des idées toutes faites, ou tâchons-nous d'approfondir notre expérience ? sommes-nous respectueux des autres, et de nous-mêmes ? savons nous écouter et comprendre celui qui parle ? Celui qui se pose ces questions-là a d'autres priorités que de satisfaire sa vanité ou de piquer dans la caisse.

16.3.5 La tentation du DSI

En 2005, les DSI français sont dans une situation difficile. La durée de vie d'un DSI dans ses fonctions est actuellement de deux ans : la moitié d'entre eux perdent leur emploi chaque année. La morosité des entreprises, leur refus d'investir, leur obsession du profit immédiat les conduisent à ne voir dans l'informatique qu'un poste de coût à comprimer. Les fournisseurs séduisent les directeurs généraux avec des arguments sommaires : « *il n'y a qu'à tout outsourcer* » (Strassmann [195]) ; « *il faut passer à l'ERP* » ; « *il vous faut un EAI* », « *il n'y a qu'à utiliser des Web Services* » etc. (dans les années 1990, « *il n'y avait qu'à passer au client/serveur* »).

C'est inquiétant. L'entreprise qui ne sait que faire de son informatique, qui tolère un tel *turn over* dans la fonction de DSI, ne peut pas maîtriser le système d'information alors qu'il est devenu son premier outil.

Confrontés à ces dangers, certains DSI cherchent à consolider leur position. Ils aspirent à accéder au comité de direction, à participer à la décision stratégique. Ils croient devoir revêtir le personnage de l'« homme de pouvoir » : parole péremptoire, répartie rapide, susceptibilité à fleur de peau, échange de bons procédés entre pairs.

Ils souhaitent, pour pouvoir parler en stratège, être celui qui définit les services que le système d'information rendra à l'entreprise et la façon dont les processus de production seront automatisés. Il faut pour cela qu'ils se mêlent de la validation des expressions des besoins, du choix des priorités, et qu'ils soient donc tout à la fois client et fournisseur, maître d'ouvrage et maître d'œuvre du système d'information.

Cette organisation convient à ceux des directeurs généraux qui pensent que « tout ça, c'est de l'informatique » et préfèrent, comme l'adjudant qui aligne une troupe, « ne voir qu'une seule tête ». Mais elle implique une confusion périlleuse entre des rôles différents.

Le DSI qui a pris la responsabilité de la maîtrise d'ouvrage, et qui a donc sous ses ordres les personnes qui établissent les expressions de besoin des divers métiers, participe en effet à deux univers mentaux différents :

- celui de la gestion de l'usine informatique, usine complexe et fragile qui doit tourner en continu sans défaillance perceptible par les utilisateurs ;

- celui de la conception du système d'information qui, elle, exige une vue prospective, une vigilance périscopique et l'aptitude à parler les divers langages de l'entreprise.

Ce DSI doit ainsi être à la fois le physicien de l'informatique et l'organisateur de l'entreprise. Or il est très difficile, et en fait humainement impossible, de conjuguer ces deux rôles : cela supposerait un écartèlement entre des préoccupations intellectuelles très différentes, une dispersion de la vigilance, et enfin des rythmes de vie et de décision inconciliables.

Certes, certains directeurs généraux croient pouvoir exiger qu'une même personne soit à la fois un saint, un héros et un génie. Mais il est dangereux de fonder une organisation sur une exigence qui ne sera pratiquement jamais satisfaite.

Dans les faits, le DSI qui ambitionne de devenir le leader effectif de la maîtrise d'ouvrage est contraint de relâcher l'attention qu'il accorde à la plate-forme technique de l'informatique. Certains d'entre eux trouveront expédient de la confier à un adjoint : alors l'organigramme de la DSI prendra la forme indiquée par la figure 16.8. Les maîtrises d'ouvrage des divers métiers seront, selon l'entreprise, rattachées hiérarchiquement soit au DSI, soit aux directeurs des métiers ; dans ce dernier cas, le DSI n'aura d'autorité directe que sur la *coordination* des maîtrises d'ouvrage, qui est une petite équipe.

Le DSI est ainsi devenu le conseiller du directeur général, l'expert qui éclaire la stratégie de l'entreprise en matière de système d'information. Le fonctionnement quotidien de l'informatique n'est plus son premier souci, c'est son adjoint qui s'en charge.

Il s'est mis ainsi dans un piège : que se passera-t-il si l'informatique connaît une défaillance grave, si par exemple une panne trop longue compromet le fonctionnement de l'entreprise ? Le directeur général cherchera, ne serait-ce que pour émettre un signal salubre, un responsable pour le punir et éventuellement le chasser de l'entreprise. Le DSI pourra-t-il dire alors « c'est mon adjoint le coupable, je n'y suis pour rien, c'est lui qui s'occupe de la plate-forme technique » ? Non, car le directeur général lui répondra

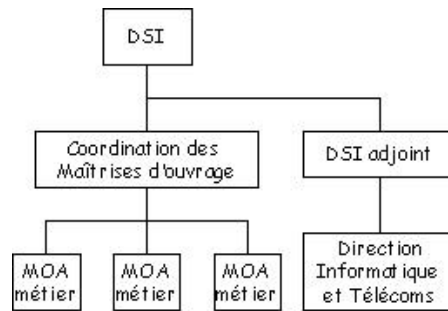


FIG. 16.8 – Un DSI devenu maître d'ouvrage

« c'est vous le patron, c'est vous le capitaine, c'est donc vous le responsable et c'est vous qui sautez ».

Ainsi le DSI qui concentre son attention sur le rôle stratégique du système d'information, qui s'érige en patron de la maîtrise d'ouvrage et délègue à un autre la responsabilité de l'usine informatique, vit avec une épée de Damoclès sur la tête : il a lâché les rênes de la plate-forme mais il sait qu'en cas d'incident grave, c'est lui qui sautera.

C'est là une situation psychologiquement et pratiquement intenable. Dans les faits ou bien l'adjoint à qui le DSI a confié l'informatique devient le véritable directeur de l'informatique, pleinement responsable, et le DSI n'est plus qu'un coordinateur de la maîtrise d'ouvrage, tâche importante mais dont les entreprises n'ont pas encore toutes reconnu la nécessité et qui ne lui donne d'autorité directe (et encore dans un rôle non hiérarchique d'animateur, de coordinateur) que sur un effectif de taille modeste par rapport à l'informatique²⁰ ; ou bien le DSI garde la main sur l'informatique, ses machines et son personnel, et alors inévitablement le poids des décisions et responsabilités que celle-ci implique tirera ses priorités du côté de la plate-forme, ce qui lui fera oublier ou négliger les aspects stratégiques du système d'information.

On risque de retrouver dans ce second cas un schéma déjà très fréquent : alors qu'il *prétend* s'occuper du système d'information (puisqu'il porte le titre de « directeur du système d'information »), le DSI ne se préoccupe en fait que de la plate-forme et se limite à jouer le rôle, d'ailleurs utile et très prenant, du physicien.

Certains DSI, jaloux de leur titre, s'emploient à empêcher l'émergence d'une compétence en maîtrise d'ouvrage car ils voient en elle un concurrent en termes de légitimité : ils ne sont pas pour rien dans la lenteur avec laquelle se met en place la maîtrise d'ouvrage professionnelle. D'autres, au contraire, souhaitent qu'une telle compétence se forme mais il leur est difficile de définir les rapports qu'ils doivent entretenir avec elle et, souvent, un conflit naîtra autour de la maîtrise du budget.

²⁰ Les effectifs consacrés à la maîtrise d'ouvrage sont de l'ordre du quart ou du cinquième de ceux de l'informatique.

Il est préférable de reconnaître dans l'entreprise deux spécialités différentes : l'une en maîtrise d'ouvrage, essentiellement fonctionnelle et sémantique, qui définit ce que le système d'information doit faire pour ses utilisateurs et pour l'entreprise ; l'autre orientée vers la physique de la plate-forme : choix des solutions d'architecture (synchronisme, persistance, concurrence), des systèmes d'exploitation, langages et SGBD, dimensionnement des ressources, reprise en cas d'incident, sauvegarde etc.

Le secret du succès, c'est d'organiser entre ces deux spécialités une dialectique mutuellement respectueuse et énergique, *sportive* en un mot, qui alimentera la dynamique de l'entreprise. Mais peu d'entreprises semblent aujourd'hui mûres pour une telle organisation. Les DSI, étant sur la défensive tout en disposant du pouvoir que donne le poids de leur direction dans le budget, sont tentés par la manœuvre désespérée et, en fait, impossible qui consiste à prendre en mains la maîtrise d'ouvrage tout en gardant le contrôle de l'informatique. Les entreprises, ne rêvant que d'oublier les soucis que cause l'informatique, sont prêtes à les écouter. Elles s'engagent ainsi dans un piège dont ni elles, ni les DSI ne pourront sortir indemnes.

16.4 La demi-stratégie

Beaucoup de nos entreprises ont non pas une stratégie, mais une moitié de stratégie. Elles veulent « grossir pour survivre », mais restent campées sur leur activité traditionnelle et refusent de diversifier leur offre. Elles veulent « réduire les coûts », mais négligent le marketing. Elles veulent « assainir les finances », mais répudient la R&D. Leurs dirigeants négligent la polyphonie de l'entreprise, la multiplicité des logiques qu'elle articule et qui toutes sont nécessaires, pour n'accorder d'attention qu'à une seule ligne mélodique.

Elles ne consacrent pas beaucoup d'efforts à l'« ingénierie d'affaires », aux « montages » qui permettraient d'offrir aux clients, sur la plate-forme traditionnelle, une gamme de services différenciés et enrichis :

- ce sont les Coréens, non les Français, qui ont introduit dans la puce du téléphone mobile les fonctions de la carte bancaire ; ce sont les Israéliens, non les Français, qui ont été le plus loin dans les services de télémédecine ;

- c'est Google, et non la Bibliothèque nationale, qui a pris l'initiative de mettre en ligne un grand nombre d'ouvrages²¹ ;

- ce sont les Américains, non les Français, qui ont transformé l'avion en un magasin où le passager peut acheter à l'occasion d'un vol des ordinateurs portables, appareils de photo numériques, CD-Rom, DVD etc. qui lui seront livrés au sol, éventuellement hors taxes et dédouanés²².

Il est vrai que l'ingénierie d'affaires suppose l'aptitude à la négociation de contrats entre égaux, aptitude rare dans des directions générales dont l'attention est absorbée par leur propre organisation interne.

21. Le service `gallica.bnf.fr` n'a jamais pris l'ampleur qu'il aurait pu et dû avoir : il a fallu que Google annonce la mise en ligne des livres pour que les politiques se réveillent.

22. Les Français en restent à la vente à bord de foulards Hermès, parfum Chanel n° 5 et autres produits « de luxe ».

Le dirigeant type « à la française » n'est pas un entrepreneur mais un gestionnaire qui a pris les rênes d'une entreprise créée par d'autres. Il s'est appliqué à prendre le pouvoir et pour le conserver il lui suffit d'être un demi-stratège.

Le milieu que forment les dirigeants « à la française » est d'une exquise complexité: la lutte pour le pouvoir sélectionne des personnes cultivées, à l'intuition rapide, à la répartie fulgurante. L'entrepreneur, le pionnier, l'innovateur y détonnent par leur concentration vers un but précis. Ils n'ont pas les bonnes manières! Les autres n'ont le plus souvent aucune peine à les éliminer.

S'ils n'y parviennent pas il leur suffira d'attendre: lorsque l'entreprise aura grossi, que les questions administratives et financières auront pris le pas sur le marketing et la technique, le pionnier s'ennuiera et partira. Alors les gestionnaires prendront les commandes, on sera revenu à la norme et l'entreprise aura perdu sa capacité à évoluer.

Je prie le lecteur de pardonner le schématisme de l'analyse ci-dessus, à laquelle on peut, et c'est heureux, opposer quelques contre-exemples. Nous avons bien sûr des entrepreneurs en France: mais comme l'Entreprise n'exerce par chez nous l'hégémonie culturelle ils ne sont ni formés, ni compris, ni promus, de sorte que leur seule pépinière est le capitalisme familial.

Pourquoi en est-il ainsi? Que l'on me permette d'être encore une fois schématique: il en est ainsi parce que, au fond, *nous autres Français ne savons pas ce que c'est que l'entreprise* car elle nous est cachée par notre héritage culturel. Ainsi nous acceptons la prétention du MEDEF, qui fédère des organisations patronales, à la représenter. Nous l'opposons à l'administration, alors que celle-ci est elle aussi une entreprise. Nous croyons que son économie est convenablement décrite par la comptabilité.

L'entreprise se trouve sur le front de taille des rapports entre l'être humain et la nature. Ces rapports sont indéfiniment perfectibles car notre connaissance de la nature n'est jamais achevée. C'est pourquoi l'image de l'entreprise stable, institutionnelle, est un faux-semblant qui sera à l'occasion bousculé non seulement par la concurrence, mais par des évolutions du rapport à la nature, des innovations auxquelles il lui faudra s'adapter.

Pourquoi aimons-nous tant ces entreprises pérennes, figées, dont les sièges sociaux imitent l'architecture d'un mausolée²³? *C'est parce que l'institution qui a, pour le meilleur et pour le pire, modelé nos valeurs et forgé notre culture, c'est l'Église, bâtie pour l'éternité sur un dogme indiscutable.*

Organisée bien avant l'État moderne, l'Église lui a fourni le modèle hiérarchique²⁴ qui s'est par la suite imposé dans nos entreprises. Nous avons ainsi appliqué à l'entreprise des méthodes qui conviennent peut-être à l'Église, mais non à une organisation orientée vers la production de choses

23. Voir les sièges sociaux des banques à Paris, boulevard des Italiens.

24. L'organisation hiérarchique n'a pas pour origine l'art de la guerre, comme en témoigne l'*Anabase* de Xénophon: des êtres humains normaux ne vont pas au combat sans se concerter et ils ne sacralisent pas l'autorité. La discipline mécanique « à la prussienne », si souvent prise pour exemple, est l'héritage des moines soldats de l'ordre Teutonique.

utiles : hiérarchie (sacralisation du pouvoir), présomption de compétence des dirigeants (comme s'ils étaient inspirés par le Saint-Esprit), liturgie des procédures (la forme prime le fond), rigidité des structures (elles-mêmes sacralisées), dogmatisme des croyances maison, sentiment d'éternité (l'apparence de la pérennité masque les risques) etc.

Pour la carrière d'un cadre, le respect de la liturgie importe alors autant ou plus que la recherche de l'efficacité. Si des hommes comme Jean-Yves Haberer ou Michel Bon ont trouvé normal de prendre la direction d'entreprises auxquelles il s'est avéré qu'ils ne pouvaient rien comprendre, c'est parce qu'ils avaient confiance en la « grâce d'état » que leur garantissait l'onction reçue lors de la cooptation parmi les dirigeants.

* *

Si notre économie n'est pas entrepreneuriale, ce n'est pas parce qu'elle est parasitée par l'État : beaucoup de grandes entreprises souffrent des travers bureaucratiques que l'on attribue à l'administration. Si nos dirigeants sont des demi-stratèges, ce n'est pas parce qu'ils sont stupides : ils ont beaucoup de talents mais ils sont coincés, tant par leur milieu que par leurs propres valeurs, dans la « position du gestionnaire ».

Nous admirons l'esprit pratique des Américains, leur tempérament de pionniers et d'entrepreneurs, la souplesse et l'organisation de leurs entreprises²⁵. Ces qualités s'expliquent par l'histoire. Leur nation, fondée par des sectes protestantes séparées de l'Église et chassées d'Europe par l'État, a accordé l'hégémonie culturelle à l'Entreprise (Gramsci [72]). Confrontée à un continent pourvu de ressources naturelles immenses, elle a mis toute son énergie dans l'organisation de l'action sur la nature en vue de produire des choses utiles.

Mais il ne nous servirait à rien de copier leurs méthodes. On ne copie que les défauts des autres quand on les imite sans faire effort sur soi-même et les Américains ne sont pas sans défauts : ils ont sans doute poussé trop loin l'exploitation de la nature et nous ferions mieux de ne pas adopter la cruelle classification en *winners* et *losers* qu'a décrite Edward Luttwak [120], ni la priorité absolue qu'ils accordent à la compétition, ni encore leur tendance à délaisser une entreprise (même utile) dès qu'elle cesse de croître.

Si nous voulons que nos entreprises soient vivantes, si nous voulons donner aux entrepreneurs l'espace qui leur est nécessaire, ce n'est pas seulement de méthodes, de procédés que nous avons besoin. Il nous faut élucider nos *valeurs*, les faire venir à la surface de notre conscience pour les examiner, les trier et les corriger ; il nous faut désacraliser le veau d'or des institutions pour réserver le sacré au seul monde des valeurs. Nous trouverons alors en nous-mêmes des ressources que nous avons négligées et nous pourrions avoir de vraies entreprises, de vraies stratégies, de vrais stratèges.

25. Elles ne sont pas parfaites (cf. Enron, Worldcom, Pan Am, et les travers des très grandes entreprises comme IBM et AT&T) ; mais elles ont un ressort qui nous surprend. Un Bill Gates n'aurait pas pu réussir en France (voir le parcours de « Guillaume Portes » dans Brulé [28] p. 307) .

Conclusion

Comment vivre avec l'automate ?

Les artefacts, dès qu'ils sortent des mains de leur producteur, transforment le monde de la nature : nos maisons, nos routes, nos ponts le rendent *habitable*. L'ubiquité de l'automate a ainsi modifié notre rapport à l'espace, sa mémoire notre rapport au temps, sa puissance les conditions du travail mental.

Il en est résulté pour nos entreprises une évolution de la fonction de production et de la fonction de coût, d'où depuis les années 1970 une transformation des formes de la concurrence. L'industrie, automatisée, n'a plus besoin d'autant de main d'œuvre. L'emploi se déploie dans les services pour assister le consommateur, désormais confronté à une offre diversifiée d'assemblages (Volle [213]).

Du coup des institutions péniblement construites au cours de notre histoire comme l'éducation, la santé, l'emploi, la justice, le syndicalisme sont frappées d'obsolescence, tout comme le mécanisme des pouvoirs législatif et exécutif. Comme elles ont sacralisé la lettre de leurs procédures il leur est difficile de revenir à l'esprit de leur mission : écrasant la bonne volonté des personnes, elles luttent autant ou même plus pour survivre que pour servir.

L'entreprise, institution décentralisée et, en outre, sans cesse renouvelée par des décès et des naissances, est le laboratoire où peut se construire le nouvel édifice institutionnel. C'est dans l'entreprise, en effet, que l'on s'approprie à l'informatique, que l'on apprend à raisonner sur les processus, à les élucider, à articuler l'être humain organisé avec l'automate programmable doué d'ubiquité, l'EHO et l'APU. À sa mission civique de toujours, qui est de produire des choses utiles²⁶, s'ajoute ainsi aujourd'hui pour l'entreprise une deuxième mission.

* *

Les décisions que cette mission réclame forment à l'horizon de la réflexion un enjeu des plus impressionnants. S'il est facile en effet d'évoquer en une

26. Certains prétendent que le mot « utilité », familier aux économistes, ne veut rien dire. Ils sont pourtant sans aucun doute attentifs à leur propre bien-être et ils protesteraient si on les privait des produits qui y contribuent. Certes le bien-être n'est pas le bonheur, mais c'est une autre question.

phrase l'articulation de l'EHO et de l'APU, la modéliser exige de démêler un écheveau complexe et suppose, avec la *pratique de l'abstraction*, une maîtrise des divers procédés de pensée que celle-ci met en œuvre.

Comme le système d'information est devenu le *langage* de l'entreprise, c'est dans l'articulation de divers types de langage que peut se trouver la solution. Les êtres humains parlent une langue naturelle riche en connotations qui éveillent des résonances dans l'esprit de l'auditeur et favorisent la communication au prix d'un risque d'ambiguïté. Le dispositif de commande de l'automate, que l'on appelle « langage de programmation », ne tolère par contre aucune ambiguïté : il est donc impropre à la communication entre des êtres humains. L'exactitude qu'il exige est plus stricte encore que celle du langage de la théorie car sa finalité est non de décrire les choses, mais de les faire fonctionner (Abelson et Sussman [197]).

Une pensée pratique, une pensée orientée vers l'action, doit savoir utiliser dans chaque étape de sa démarche celui de ces trois langages qui convient le mieux : elle doit être *polyglotte*, ce mot désignant ici non la maîtrise de plusieurs langues naturelles mais celle, en profondeur, de couches différentes du langage.

Apprendre le langage de la théorie - donc, en tout premier, celui des mathématiques - est une épreuve pour les esprits les plus fins que cela contraint à renoncer à la puissance suggestive du langage naturel. Apprendre pour la première fois un langage de programmation est tout aussi pénible car il faut s'accoutumer à une nouvelle façon de raisonner. Mais alors c'est un continent qui s'ouvre : si les langages de programmation sont tous en principe équivalents chacun correspond en effet à une intuition, à une vue sur le monde différente. Une fois assimilé l'arbitraire des notations on découvre la diversité de ces intuitions dans LISP et Prolog, dans Fortran, Cobol, C++ et Java, dans Perl et Python etc. Le monde des programmes est simple sans doute, mais pas plus simple à coup sûr que celui des nombres entiers dont la théorie, on le sait, ne sera jamais achevée.

Après les langages viennent les algorithmes dont la compréhension ne se sépare pas de celle de leur réalisation physique par l'automate (Knuth [105]), puis les systèmes d'exploitation et les architectures qui sont à l'art de la programmation ce que les plus hautes prouesses de l'ingénierie sont à l'art de la mécanique. Langages, algorithmes, systèmes d'exploitation et architectures forment l'objet de la science informatique et de la compétence professionnelle des informaticiens (Printz [168] et Bloch [14]).

Si n'importe qui peut jouer en virtuose sur un écran-clavier après quelques mois d'apprentissage, il faut pour comprendre les enjeux de l'informatique une autre maturité, une autre concentration, une autre modestie. Les penseurs qui s'y attelleront devront accepter les souffrances de l'apprentissage, les lenteurs de l'expérimentation, la contradiction des faits, l'humiliation du doute et la frustration de l'inachevé.

* *

L'automate est-il *bon* ou *mauvais*? Question futile : l'automate est comme le marteau qui peut servir à planter des clous comme à assommer son pro-

chain. Le bien ou le mal résident dans l'intention, non dans l'outil. Mais comme l'automate change le monde, comme il déplace la frontière du possible, il invite à poser des questions que nous pouvions croire réglées depuis longtemps : que voulons-nous *être*, quel sens donnons-nous à notre humanité ? Que voulons-nous *faire* du monde que notre action transforme ? Et, pour commencer comme il se doit à petite échelle, que voulons-nous faire de nos entreprises ?

Leur valeur suprême, c'est l'*efficacité* : il s'agit de faire au mieux, de produire le plus d'utilité possible, avec les ressources dont elles disposent. Mais l'automate modifie la façon dont s'incarne l'efficacité et, pour anticiper l'avenir de celle-ci, il est utile d'examiner sa généalogie.

Notre économie, nos entreprises, sont en tant qu'institutions nées avec l'industrie dans l'Angleterre du XVIII^e siècle. L'économie industrielle s'est bâtie sur l'*échange équilibré* qui seul pouvait procurer la sécurité et les débouchés nécessaires à la rentabilisation de ses machines. Celui qui achète et celui qui vend, également libres de leurs transactions, se rencontrent sur le marché : l'équilibre de l'échange suppose non seulement que les valeurs des biens échangés soient égales mais aussi qu'aucune des parties n'ait le pouvoir de contraindre l'autre.

L'économie agricole et féodale antérieure (Bloch [12]) s'était appuyée sur l'échange inégal, sur une *prédation* qu'équilibrait la *charité* : celui qui s'était emparé de la richesse à la pointe de l'épée devait subvenir, en principe, aux besoins des pauvres. C'est pour polémiquer contre le couple de la prédation et de la charité, pour faire l'éloge de l'échange équilibré qu'Adam Smith [191], le génial fondateur de la science économique, a évoqué la *main invisible* qui guiderait la société vers un optimum que personne n'a explicitement voulu²⁷. Si on lit les passages où il critique l'égoïsme des négociants, on comprend qu'il n'a jamais entendu faire l'apologie de la prédation.

Dans l'économie réelle où, des plus archaïques au plus récentes, les formes d'organisation se superposent comme des couches géologiques, la prédation n'a bien sûr jamais disparu : les prédateurs sont à l'affût et tirent profit, c'est le cas de le dire, de toutes les occasions en particulier sur le marché du travail ou sur celui des matières premières (Verschave [208])²⁸. Le combat qu'Adam Smith a engagé pour l'échange équilibré a donc dû, devra encore être poursuivi avec persévérance, mais c'est lui qui, associé à la division du travail, a orienté nos institutions et nos lois.

27. "Every individual necessarily labours to render the annual revenue of the society as great as he can. He generally neither intends to promote the public interest, nor knows how much he is promoting it. By preferring the support of domestic to that of foreign industry, he intends only his own security; and by directing that industry in such a manner as its produce may be of the greatest value, he intends only his own gain, and he is in this, as in many other cases, led by an invisible hand to promote an end which was no part of his intention. Nor is it always the worse for society that it was no part of his intention. By pursuing his own interest he frequently promotes that of the society more effectually than when he really intends to promote it. I have never known much good done by those who affected to trade for the public good." (Adam Smith, *An Inquiry into the Nature and Causes of the Wealth of Nations* 1776).

28. www.volle.com/lectures/verschave.htm.

Cependant l'automate, en modifiant le contenu de la division du travail, exige un nouvel équilibre. Alors que la main d'œuvre industrielle était spécialisée dans des opérations *physiques*, l'agent opérationnel de l'entreprise automatisée est spécialisé dans des opérations *mentales*. À chaque spécialité correspond ainsi un *métier* dont l'univers mental est spécifique. Dès lors l'entreprise court le risque d'un éclatement en spécialités mutuellement étanches, incapables de se comprendre et de dialoguer. Or l'efficacité exige que les métiers coopèrent dans la réalisation du processus de production, puis avec les fournisseurs et partenaires, enfin avec les clients eux-mêmes car dans une économie diversifiée la réponse au besoin suppose un dialogue avec le client.

C'est là sans doute la meilleure interprétation que l'on puisse donner de la *société de l'information* : pour pouvoir coopérer, il faut se comprendre ; il faut savoir écouter ce que dit l'autre et donc le *respecter*. L'équilibre de l'échange, jadis limité au marché des produits, s'étend maintenant au *commerce de la considération*.

Il s'agit non de bons sentiments ni d'injonction morale mais d'une obligation pratique, d'une contrainte qu'implique immédiatement la recherche de l'efficacité. Si celle-ci rejoint ainsi des valeurs humaines, qui pourra s'en plaindre ?

* *

Ceux à qui répugne, à juste titre, l'ornière des bons sentiments tombent parfois, sous prétexte de réalisme, dans celle du cynisme : certains se réfèrent ainsi à Darwin pour justifier, dans les entreprises, un sacrifice humain censé contribuer à la promotion des plus capables. Lorsqu'on évoque, avec une admiration gourmande, le caractère de ces « tueurs » et « tueuses » dont « les dents rayent le parquet », on encourage des comportements nuisibles. Notre esthétique enfin accorde une place prédominante au spectacle de la violence, comme si nous avions la nostalgie de la prédation. Nous n'avons pas fait le ménage dans l'édifice des valeurs.

Au combat contre la prédation et pour l'échange équilibré doit cependant désormais s'associer le combat pour l'*équilibre de la considération*. Outre le bien-être des populations, l'enjeu réside dans le choix entre les perspectives également ouvertes, également possibles, de la civilisation et de la barbarie : l'automate, neutre par lui-même, peut en effet mettre sa puissance au service de l'une comme de l'autre.

Il est facile, si l'on réfléchit un tant soit peu, d'entrevoir les horizons sur lesquels débouchent ces perspectives. *Apprendre à vivre avec l'automate* apparaît alors comme un objectif que chacun doit poursuivre pour son propre compte et pour lequel il convient de militer d'abord dans nos entreprises, puis au niveau de la société entière.

Bibliographie

- [1] *Perspectives of the Founding Fathers*, Winter Simulation Conference, 1992.
- [2] Norman Abramson. The aloha system—another alternative for computer communications. In *AFIPS Conference Proceedings*, volume 37. AFIPS, 1970.
- [3] Jean-Pierre Adam. *La construction romaine, matériaux et techniques*. Picard, 2005.
- [4] Ravi Sethi et Jeffrey Ullman Alfred Aho. *Compilers*. Addison Wesley, 1988.
- [5] Masahiko Aoki. Information and governance in the silicon valley model, 1999.
Dans la Silicon Valley, les animateurs du capital risque ont une excellente expertise technique et en font bénéficier les entreprises dans lesquelles ils investissent (www-econ.stanford.edu/faculty/workp/swp99028.pdf).
- [6] Jacques Arsac. *Préceptes pour programmer*. Dunod, 1991.
Les conseils d'un sage.
- [7] Gaston Bachelard. *Le rationalisme appliqué*. PUF, 1949.
- [8] John Backus. The history of fortran. In Richard Wexelblat, editor, *History of Programming Language*. ACM, Academic Press, 1978.
Les concepteurs du langage Fortran ont eu, entre autres préoccupations, le souci d'optimiser la performance de la machine.
- [9] Robert-Vincent Joule et Jean-Léon Beauvais. *Petit traité de manipulation à l'usage des honnêtes gens*. Presses Universitaires de Grenoble, 2002.
- [10] David S. Bennahum. *Extra Life*. Basic Books, 1998.
Pour toute une génération, l'apprentissage des jeux sur ordinateur a été la porte d'entrée vers l'informatique.
Voir www.volle.com/lectures/bennahum.htm.
- [11] Tim Berners-Lee. *Weaving the Web*. Harper Business.
L'histoire de la Toile par celui qui l'a créée en 1991.
- [12] Marc Bloch. *La société féodale*. Albin Michel, 1939.
Une forme d'organisation que l'on retrouve encore aujourd'hui.
Voir www.volle.com/lectures/bloch.htm.
- [13] Laurent Bloch. *Initiation à la programmation avec Scheme*. Technip, 2001.

- Introduction au langage qui procure le plus de plaisir au programmeur.
Voir www.laurent-bloch.org/Livre-Scheme/TDM.html.
- [14] Laurent Bloch. *Les systèmes d'exploitation des ordinateurs : histoire, fonctionnement, enjeux*. Vuibert, 2003.
Pour découvrir les diverses fonctions que remplit le système d'exploitation.
Voir www.laurent-bloch.org/Livre-Systeme/index.html.
- [15] Laurent Bloch. *Systèmes d'information, obstacles et succès*. Vuibert, 2005.
Une réflexion philosophique et pratique sur la construction et le rôle du système d'information.
Voir www.laurent-bloch.org/SI-Projets-extraits/index.html.
- [16] Maurice Blondel. *L'Action*. Félix Alcan, 1893.
L'œuvre maîtresse du philosophe qui, fait exceptionnel, a mis l'*action* au point de départ de sa pensée.
- [17] Nicolas Boileau-Despréaux. *L'art poétique*, 1674.
- [18] L. Bonfante. *La naissance des écritures*. Seuil, 1994.
- [19] Grady Booch, James Rumbaugh et Ivar Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley, 1998.
- [20] N. Bourbaki *Éléments de mathématique, Théorie des ensembles*. Hermann, 1966.
- [21] N. Bourbaki *Éléments d'histoire des Mathématiques*. Hermann, 1969.
- [22] Pierre Bourdieu. Éléments d'une théorie sociologique de la perception artistique. *Revue internationale des sciences sociales*, 1968.
- [23] Jacques Bouveresse. *Prodiges et vertiges de l'analogie*. Raisons d'agir, 1999.
Bouveresse prouve, par ce livre, que l'on peut philosopher sans céder aux prestiges de la mode ni renoncer en rien à la rigueur.
- [24] Nicolas Bouzou. Le commerce écrase les prix. *Xerfi Previsis*, (90), juin 2005.
- [25] Isabelle Boydens. *Informatique, normes et temps*. Bruylant, 1999.
À partir du cas particulier de la base de données de la sécurité sociale belge, Isabelle Boydens analyse les problèmes que pose la pratique des bases de données et propose des solutions.
Voir le commentaire page 81.
- [26] Philippe Breton. *Une histoire de l'informatique*. La Découverte, 1987.
- [27] Dan Bricklin. Software arts and visicalc.
www.j-walk.com/ss/history/spreadsh.htm.
- [28] Jean-Pierre Brulé. *L'informatique malade de l'Etat*. Les Belles Lettres, 1993.
Le constat amer d'un échec, par un ancien PDG de Bull.
- [29] Martin Campbell-Kelly. *Une histoire de l'industrie du logiciel*. Vuibert Informatique, 2003.
- [30] S. Syme et L. J. Camp Code as governance, the governance as code. *Social Science Research Network Electronic Paper Collection*, avril 2001.
- [31] René Carmille. *La mécanographie dans les administrations*. Recueil Sirey, 1942.

- [32] Nicholas G. Carr. IT does'nt matter. *Harvard Business Review*, mai 2003.
Une attaque en règle contre l'informatique - mais elle rate sa cible.
- [33] Paul Carroll. *Big Blues, The Unmaking of IBM*. Crown, 1994.
Une histoire détaillée de la chute de la maison IBM dans les années 1990... mais IBM s'est redressée par la suite, ce qui infirme quelque peu l'analyse.
- [34] Mary Carruthers. *Machina memorialis*. Gallimard, 2002.
- [35] Paul E. Ceruzzi. *A History of Modern Computing*. MIT, 1998.
- [36] Michael Hammer et James Champy. *Re-engineering the Corporation*. Harper Business, 1993.
Aux origines du *Business Process Reengineering*.
- [37] Carl von Clausewitz. *Vom Kriege*, 1832.
Un modèle pour quiconque s'efforcera de faire le tour d'une question complexe.
- [38] Markus Clermont. *A Scalable Approach to Spreadsheet Visualization*. PhD thesis, Universität Klagenfurt, mars 2003.
- [39] Robert P. Colwell. *The Pentium Chronicles*. Wiley & Sons, 2006.
- [40] Michael Crichton. *Disclosure*. Knopf, 1994.
- [41] Michael Crichton. *Airframe*. Knopf, 1996.
Voir www.volle.com/lectures/airframe.htm.
- [42] François Dagognet. *L'invention de notre monde*. Encre Marine, 2000.
- [43] Kristen Nygaard et Ole-Johan Dahl. The development of the simula language. In Richard Wexelblat, editor, *History of Programming Language*. ACM, Academic Press, 1978.
- [44] O. J. Dahl, E. W. Dijkstra et C. A. R. Hoare. *Structured Programming*. Academic Press, 1972.
- [45] Magali Demotes-Mainard. La connaissance statistique de l'immatériel. *Groupe de Voorburg sur la statistique des services*, 2003.
- [46] René Descartes. *Discours de la Méthode*, 1637.
- [47] Vincent Desportes. *L'Amérique en armes*. Economica, 2002.
Présentation de deux modèles qui schématisent utilement le rôle de l'automate dans l'action.
- [48] Alain Le Diberder. *Histoire d'@*. La Découverte, 2000.
www.clve.fr/sabir_cyber.htm.
- [49] Gilles Dostaler. *Keynes et ses combats*. Albin Michel, 2005.
Voir www.volle.com/lectures/airframe.htm.
- [50] Gilles Dowek. A propos de quelques démonstrations pas très convaincantes. Technical report, INRIA, 2001.
- [51] Gérard Dréan. Apple, sun et... ibm. *Le Monde Informatique*, 22 mars 1996.
- [52] Ivar Ekeland. *Éléments d'économie mathématique*. Hermann, 1979.
Une présentation rigoureuse et claire de la nature et des conditions de l'équilibre économique.
- [53] Danielle Elisseeff. *Confucius, des mots en action*. Gallimard, 2003.
Voir www.volle.com/lectures/elisseeff.htm.
- [54] Robert Escarpit. *L'information et la communication. Théorie générale*. Hachette, Paris, 1991.

- [55] Jeanne Favret-Saada. *Les mots, la mort, les sorts*. Gallimard, 1985.
Si la magie est la manipulation des symboles en vue d'obtenir du pouvoir, ne se pratique-t-elle pas aussi dans nos entreprises ?
Voir www.volle.com/lectures/favret.htm.
- [56] Daniel Friedman et Matthias Felleisen. *The Little Schemer*. MIT Press, 2000.
La présentation pédagogique de ce livre est astucieuse, mais il réclame une lecture lente et répétée.
Voir www.volle.com/lectures/littleschemer.htm.
- [57] Matthias Felleisen, Robert Bruce Findler, Matthew Flatt et Shriram Krishnamurthi. *How to Design Programs*. MIT Press, 2001.
- [58] Daniel Fixari. Le calcul économique, ou de l'utilisation des modèles irréalistes. *Annales des Mines*, avril 1977.
Pour qu'un modèle soit utile, il n'est pas nécessaire qu'il soit réaliste.
- [59] Terry Winograd et Fernando Flores. *Understanding computers and cognition*. Addison-Wesley, 1989.
- [60] George Forsythe. Computer science and education. In *Information processing 68*. North-Holland, 1969.
- [61] Michel Foucault. *Les mots et les choses*. Gallimard, 1966.
- [62] Bill Gates. An open letter to lobbyists. *Computer Notes*, février 1976.
La lettre qui a inauguré le marché des logiciels pour micro-ordinateur.
- [63] Kurt Gödel. Über formal unentscheidbare sätze der principia mathematica und verwandter systeme. *Monatshefte für Mathematik und Physik*, (38), 1931.
L'article où est démontré le théorème de Gödel.
- [64] Michel Gensollen. Économie des biens informationnels. In « *Autour du Libre* », colloque de l'Institut National des Télécommunications, Evry, mai 2002.
- [65] Louis V. Gerstner. *Whos Says Elephants Can't Dance ?* Harper Business, 2002.
Le redressement d'IBM.
- [66] Bertrand Gille. *Histoire des techniques*. Gallimard La Pléiade, Paris, 1978.
La modélisation de l'histoire sous la forme d'une succession de systèmes techniques.
Voir www.volle.com/lectures/gille.htm.
- [67] Étienne Gilson. *Le thomisme*. Librairie philosophique J. Vrin, Paris, quatrième édition, 1944.
- [68] Étienne Gilson. *L'être et l'essence*. Librairie philosophique J. Vrin, Paris, 1948.
- [69] James Gleick. *Chaos: Making a new Science*. Penguin, 1988.
- [70] James Gleick. *Genius : the Life and Science of Richard Feynman*. Vintage Books, 1992.
- [71] Robert Gordon. Has the "new economy" rendered the productivity slowdown obsolete? Technical report, juin 1999.
Dans la lignée de Solow, le travail d'un sceptique.
- [72] Antonio Gramsci. *Note sul Machiavelli, Americanismo e fordismo*. Istituto Gramsci, 1975.

- [73] Jean Grandmont. *Money and Value*. Cambridge University Press, 1983.
- [74] John Grisham. *The Firm*. Doubleday, 1991.
- [75] Georg Groddeck. *Das Buch vom Es*, 1923.
- [76] Andrew S. Grove. *Only the Paranoid survive*. Doubleday, 1996.
- [77] Bernard Guibert, Jean Laganier et Michel Volle. Essai sur les nomenclatures industrielles. *Économie et Statistique*, (20), février 1971.
www.volle.com/articles/nomenclature.htm.
- [78] Guillaume Lecointre et Hervé Le Guyader. *Classification phylogénétique du vivant*. Belin, 2001.
- [79] Amy Harmon. Internet gives teenage bullies weapons to wound from afar. *The New York Times*, 26 août 2004.
- [80] Armand Hatchuel. Repenser la gestion : un point de vue historique sur les firmes comme innovations institutionnelles. *La lettre de la régulation*, (47), janvier 2004.
web.upmf-grenoble.fr/regulation/Lettre_regulation/lettrepdf/LR47.pdf.
- [81] J. S. Henry. The impact of lotus notes on productivity. Technical report, 1992.
- [82] Michael Hiltzik. *Dealers of Lightning - Xerox PARC and the Dawn of the Computer Age*. Harper Business, 1999.
Histoire du centre de recherche où furent mises au point, dans les années 1970, plusieurs innovations essentielles.
Voir www.volle.com/lectures/hiltzik.htm.
- [83] Karim K. Hirji. Exploring data mining implementation. *Communications of the ACM*, juillet 2001.
- [84] Andrew Hodges. *Alan Turing: The Enigma*. Walker & Company, 2000.
Biographie d'Alan Turing.
- [85] Grace Hopper. Keynote address. In Richard Wexelblat, editor, *History of Programming Language*. ACM, Academic Press, 1978.
Comment le besoin d'un langage de programmation a émergé au début des années 50.
- [86] Watts S. Humphrey. Software unbundling : A personal perspective. *IEEE Annals of the History of Computing*, janvier-mars 2002.
- [87] Edmund Husserl. *Méditations cartésiennes*. Librairie philosophique J. Vrin, 1929.
- [88] Georges Ifrah. *Histoire universelle des chiffres*. Robert Laffont, 1994.
Voir www.volle.com/lectures/ifrah.htm.
- [89] Michael D. Intriligator. *Mathematical Optimization and Economic Theory*. Prentice-Hall, 1971.
- [90] Jean-Marc Jancovici. *L'avenir climatique*. Seuil, 2002.
- [91] Gérard Jean. *Urbanisation du business et des SI*. Hermès, 2000.
Voir www.volle.com/lectures/jean.htm.
- [92] Antoine-Henri Jomini. *Précis de l'art de la guerre*, 1838.
- [93] François Jullien. *La propension des choses*. Seuil, 1992.

- [94] François Jullien. *Zhong Yong, la régulation à usage ordinaire*. Imprimerie nationale, 1993.
- [95] François Jullien. *Un sage est sans idée*. Seuil, 1998.
- [96] V. G. Cerf et R. E. Kahn. A protocol for packet network intercommunication. *IEEE Trans. Comm. Tech.*, mai 1974.
La note technique à l'origine du protocole TCP/IP.
- [97] Larry R. Squire et Eric R. Kandel. *Memory: From Mind to Molecules*. Scientific American Library, 1999.
- [98] Alan C. Kay. The early history of smalltalk. In Thomas Bergin et Richard Gibson, editor, *History of Programming Languages II*. Addison-Wesley, 1996.
De la découverte de Simula à la conception de Smalltalk, l'invention du premier langage de programmation orienté objet.
- [99] Peter G. W. Keen. *Shaping the future. Business design through information technology*. Harvard Business School Press, 1991.
Ce livre sur la modélisation des processus a eu une grande influence dans les années 90.
- [100] William Kent. *Data and Reality. Basic Assumption in Data Processing Reconsidered*. Elsevier North-Holland, 1981.
- [101] Tracy Kidder. *The Soul of a New Machine*. Atlantic-Little, Brown, 1981.
Le meilleur récit que je connaisse sur la conception et la construction d'un ordinateur.
Voir le commentaire page 48.
- [102] Søren Kierkegaard. *Post-Scriptum définitif et non scientifique aux Miettes Philosophiques*. Editions de l'Orante, 1977.
- [103] Donald E. Knuth. Computer programming and computer science. In *Dictionary of Science and Technology*. Academic Press, 1992.
- [104] Donald E. Knuth. *Selected Papers on Computer Science*. CSLI, 1996.
- [105] Donald E. Knuth. *The Art of Computer Programming*. Addison Wesley, 1997.
Ce monumental traité d'algorithmique est l'un des textes fondamentaux en informatique.
Voir www.volle.com/lectures/knuth.htm.
- [106] Alfred Korzybski. *Une carte n'est pas le territoire*. L'Eclat, 1998.
- [107] Vincent S. Lai. Intraorganizational communication with intranets. *Communications of the ACM*, juillet 2001.
- [108] André Lalande. *Vocabulaire technique et critique de la philosophie*. PUF, 1976.
- [109] Thomas K. Landauer. *The Trouble with Computers*. MIT Press, 1996.
Les ordinateurs poseraient plus de problèmes qu'ils n'apportent de solutions... après Solow et Gordon, un autre sceptique.
- [110] Pierre Legendre. *L'empire de la Vérité*. Arthème Fayard, 1983.
- [111] Yeshayahou Leibowitz. *Israël et le judaïsme*. Desclée de Brouwer, 1996.

- [112] Jean Tirole et Josh Lerner. The simple economics of open source. *Journal of Industrial Economics*, (52), juillet 2001.
- [113] Steven Levy. *Hackers*. Delta Publishing, 1994.
Les rêves et l'aventure des pionniers qui ont inventé l'informatique d'aujourd'hui.
Voir www.volle.com/lectures/hackers.htm.
- [114] Steven Levy. *Crypto*. Viking, 2001.
- [115] J. C. R. Licklider. Man computer symbiosis. *IRE Transactions on Human Factors in Electronics*, mars 1960.
Un texte dont nous n'avons pas encore épuisé la profondeur.
- [116] J. C. R. Licklider. The computer as a communication device. *Science and Technology*, avril 1968.
Un texte prophétique, si l'on pense à la date de sa publication.
- [117] L. Landau et E. Lifchitz. *Mécanique*. MIR, 1966.
- [118] Robert Ligonnière. *Préhistoire et histoire des ordinateurs*. Robert Laffont, 1987.
- [119] Christophe Longépé. *Le projet d'urbanisation du SI*. Dunod, 2004.
Un excellent manuel pour ceux qui doivent urbaniser un système d'information.
- [120] Edward N. Luttwak. *Turbo Capitalism*. Harper Collins, 1999.
Une analyse lucide de l'économie américaine.
Voir www.volle.com/lectures/luttwak.htm.
- [121] Pierre Lévy. *L'intelligence collective et ses objets*. La Découverte, 1994.
- [122] Pierre Lévy. *World Philosophy*. Odile Jacob, 2000.
- [123] Katie Hafner et Matthew Lyon. *Where Wizards Stay Up Late*. Touchstone, 1998.
Une excellente histoire de l'Internet.
- [124] Michael Malone. *The Microprocessor, a Bibliography*. Springer-Verlag, 1995.
Pour comprendre comment le micro-processeur est né, comment il est fabriqué et comment il fonctionne.
- [125] René Mandel. *De la stratégie business aux systèmes d'information*. Lavoisier, 2006.
De la modélisation de l'entreprise à l'alignement stratégique du système d'information.
- [126] Richard Stiller et Jos Marlowe. *Asunder*. Briarwood, 1999.
Récit des origines de Java.
- [127] Richard Mattessich. Spreadsheet: Its first computerization (1961-1964).
www.bricklin.com/history/sai.htm.
- [128] Marshall McLuhan. *Understanding Media : The Extensions of Man*. MIT Press, 1964.
- [129] Elliotte Rusty Harold et W. Scott Means. *XML in a Nutshell*. O'Reilly, 2004.
- [130] Jean-Pierre Meinadier. *Le métier d'intégration de système*. Hermès, 2002.
L'ingénierie des systèmes d'information est un cas particulier de l'ingénierie de système, et peut s'inspirer utilement des acquis de cette discipline.

- [131] Sally Shlaer et Stephen J. Mellor. *Object Lifecycles: Modeling the World in States*. Yourdon, 1991.
Concevoir l'objet dans son cycle de vie.
- [132] Bertrand Meyer. *Conception et programmation par objets*. InterEditions, 1990.
- [133] Stan Davis et Christopher Meyer. *BLUR: The speed of change in the connected economy*. Addison-Wesley, 1998.
- [134] Claude Mignot. *L'architecture du XIX^e siècle*. Office du Livre, 1983.
Pour comprendre l'esthétique de l'économie industrielle, il faut regarder les bâtiments qu'elle a conçus.
- [135] Simon Nora et Alain Minc. *L'informatisation de la société*. Documentation française, 1978.
- [136] Jacques Mélése. *L'analyse modulaire des systèmes de gestion*. AMS, Hommes et Techniques, 1972.
- [137] Audris Mockus, Roy Fielding et James Herbsleb. « case study of open source software program movement: The apache server », 2000.
- [138] Jean-Louis Le Moigne. *Le constructivisme*. ESF, 1980.
- [139] Simon Sebag Montefiore. *Stalin The Court of the Red Tsar*. Orion, 2003.
- [140] Gordon E. Moore. Cramming more components into integrated circuits. *Electronics*, avril 1965.
L'article à l'origine de la « Loi de Moore ».
- [141] Charles Mopsik. *La Cabale*. Grancher, 1988.
Voir www.volle.com/lectures/mopsik.htm.
- [142] Edgar Morin. *Science et conscience de la complexité*. Librairie de l'Université, Aix en Provence, 1984.
- [143] Pierre Mounier-Kuhn. *L'Informatique en France de la Deuxième Guerre mondiale au Plan Calcul : science, industrie, politiques publiques*. PhD thesis, CNAM, 1999.
- [144] Ramez Elmasri et Shamkant B. Navathe. *Fundamentals of Database Systems*. Addison Wesley, quatrième édition, 2003.
Ouvrage de référence sur les bases de données.
- [145] Joseph Needham. *Science and Civilisation in China*. Cambridge University Press, 1962.
Voir www.volle.com/lectures/needham.htm.
- [146] John von Neumann. *First Draft of a Report on the EDVAC*. Technical report, Moore School of Electrical Engineering, University of Pennsylvania, 1945.
- [147] John von Neumann. *The Computer and the Brain*. Yale Nota Bene, 2000.
Un petit livre fondamental, dont la rédaction a été malheureusement interrompue par la mort de l'auteur, sur les relations entre le cerveau et l'ordinateur.
- [148] Patrick O'Beirne. *Spreadsheet Check and Control*. Systems Publishing, 2003.
Contrôler la qualité des tableurs que l'entreprise utilise.
- [149] Jean Orioux. *Talleyrand*. Flammarion, 1998.

- [150] Ray Panko. Spreadsheet research.
panko.cba.hawaii.edu/SSR/home.htm.
- [151] Blaise Pascal. *De l'esprit géométrique et de l'art de persuader*, 1655.
- [152] Blaise Pascal. *Pensées (in Œuvres complètes)*. Gallimard, 1998.
- [153] John L. Hennessy et David A. Patterson. *Architecture des ordinateurs*. Vuibert Informatique, 2003.
Ce gros manuel est une mine d'information.
- [154] Richard E. Pattis. *Karel the Robot*. Wiley, 1995.
Un petit manuel qui permet de bien comprendre ce qui se passe entre le programmeur et l'ordinateur.
- [155] Linda Dailey Paulson. Wanted: More network-security graduates and research. *Computer*, février 2002.
- [156] Jean-Louis Peaucelle. *Informatique rentable et mesure des gains*. Hermès, 1997.
Un bon ouvrage sur l'économie du système d'information.
- [157] Jean-Louis Peaucelle. *Systèmes d'information : Le point de vue des gestionnaires*. Economica, 1999.
Tout ce que doivent savoir les dirigeants pour comprendre leur système d'information.
- [158] Pete Peterson. *Almost Perfect*. Prima Publishing, 1994.
Histoire de WordPerfect.
- [159] Thomas Petzinger. *Hard Landing*. Times Business, 1995.
Cette histoire du transport aérien contient de bons et de mauvais exemples en matière de stratégie d'entreprise.
Voir www.volle.com/lectures/Petzinger.htm.
- [160] Edmond Marc et Dominique Picard. *L'école de Palo Alto*. Retz, 1984.
Comment pratiquer la manipulation pour la bonne cause.
Voir www.volle.com/lectures/paloalto.htm.
- [161] René-Victor Pilhes. *L'imprécatteur*. Seuil, 1974.
- [162] Raymond Poincaré. *La valeur de la science*, 1905.
- [163] Karl Popper. *Objective Knowledge*. Oxford University Press, 1979.
- [164] Karl Popper. *The Logic of Scientific Discovery*. Routledge, 2002.
Le livre qui a, pour la première fois, fourni le critère de la scientificité.
- [165] Jacques Printz. *Puissance et limites des systèmes informatisés*. Hermes, 2000.
Pour bien utiliser l'informatique, il faut concevoir ce qu'elle peut faire, et aussi savoir qu'elle ne peut pas faire.
- [166] Jacques Printz. *Coûts et durée des projets informatiques*. Hermes, 2001.
Une référence sur l'évaluation des coûts, depuis les règles de pouce utile jusqu'aux méthodes élaborées.
- [167] Jacques Printz. *Productivité des programmeurs*. Hermes, 2001.
Mettre davantage de programmeurs sur un projet en retard, cela ne fera qu'accroître le retard...
- [168] Jacques Printz. *Le génie logiciel*. PUF, 2005.
Un Que sais-je ? sur l'ingénierie du système d'information.

- [169] Philippe Quéau. *Le virtuel : Vertus et Vertiges*. Champ Vallon, 1993.
Pourrions-nous un jour nous rencontrer dans une salle de réunion virtuelle, où chacun de nous sera représenté par son image en 3D ?
- [170] John Rawls. *A Theory of Justice*, 1971.
- [171] Eric S. Raymond. The cathedral and the bazaar.
Un des textes les plus célèbres du mouvement « open source »
www.catb.org/~esr/writings/cathedral-bazaar/.
- [172] Eric S. Raymond. *The Art of Unix Programming*. Addison Wesley, 2004.
Description vivante du monde de la programmation autour d'Unix, de ses intentions, de son histoire, des problèmes qu'il rencontre, des outils dont il dispose.
- [173] D. Rehr. The typewriter. *Popular Mechanics*, août 1996.
- [174] L. G. Roberts. Aloha packet system with and without slots and capture. Technical Report 8, Stanford Research Institute, 1972.
- [175] Paul Romer. Endogeneous technical change. *Journal of Political Economy*, 1990.
- [176] Winn L. Rosch. *Hardware Bible*. SAMS Publishing, 1997.
- [177] Benjamin M. Rosen. Visicalc : Breaking the personal computer software bottleneck. *Morgan Stanley Electronics Letter*, 11 juillet 1979.
L'article qui a inauguré le phénomène de l'informatique personnelle.
- [178] Jean E. Sammet. *Programming Languages : History and Fundamentals*. Prentice-Hall, 1969.
- [179] Ferdinand de Saussure. *Cours de linguistique générale*. Payot, Paris, 1916.
Un texte fondamental pour toute réflexion sur le langage.
Voir www.volle.com/lectures/saussure.htm.
- [180] Bruce Schneier. Semantic network attacks. *Communications of the ACM*, décembre 2000.
- [181] Martin Fowler et Kendall Scott. *UML distilled second edition*. Addison-Wesley, 1999.
- [182] Ravi Sethi. *Programming Languages*. Addison Wesley, 1997.
- [183] Lucien Sfez. *La décision*. PUF, 2004.
- [184] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, juillet - octobre 1948.
- [185] Claude E. Shannon. *The Mathematical Theory of Communication*. University of Illinois Press, 1963.
- [186] Herbert A. Simon. *The Sciences of the Artificial*. MIT Press, 1969.
- [187] Simon Singh. *Fermat's Enigma*. Anchor Books, 1997.
Une intéressante introduction à la théorie des nombres.
- [188] Simon Singh. *The Code Book*. Anchor Books, 1999.
La théorie des nombres appliquée au chiffrement.
Voir www.volle.com/lectures/singh.htm.
- [189] Michael Sipser. *Introduction to the Theory of Computation*. PWS, 1997.
Un petit chef d'œuvre de pédagogie.
Voir www.volle.com/lectures/sipser.htm.

- [190] D. Sjobbema. *Ils ont inventé l'électronique*. Publitrone/Elektor, 1999.
- [191] Adam Smith. *An Inquiry into the Nature and Causes of the Wealth of Nations*. Methuen and Co, 1776.
- [192] Robert Solow. A contribution to the theory of economic growth. *Quarterly Journal of Economics*, 1956.
- [193] Richard Stallman. The gnu operating system and the free software movement. In Sam Ockman et Mark Stone Chris DiBona, editor, *Open Sources: Voices from the Open Source Revolution*. O'Reilly, 1999.
- [194] Adin Steinsaltz. *La Rose aux treize pétales*. Albin Michel, 1989.
- [195] Paul A. Strassmann. Outsourcing: A game for losers. *Computerworld*, août 1995.
Cet article date, et l'outsourcing a évolué depuis, mais il dénonce des illusions qui sont encore vivaces.
- [196] Bjarne Stroustrup. *The Design and Evolution of C++*. Addison Wesley, 1999.
Les intentions sous-jacentes à la conception d'un langage.
- [197] Harold Abelson et Gerald Jay Sussman. *Structure and Interpretation of Computer Programs*. MIT Press, 1996.
Une des meilleures introductions à l'art de la programmation.
Voir www.volle.com/lectures/Scheme.htm.
- [198] Andrew Tanenbaum. *Structured Computer Organization*. Prentice-Hall, 1984.
- [199] Andrew Tanenbaum. *Operating Systems, Design and Implementation*. Prentice-Hall, 1987.
Le livre qui est à l'origine de Linux.
- [200] Andrew Tanenbaum. *Computer Networks*. Prentice-Hall, 1989.
- [201] Olivier Marchand et Claude Thélot. *Deux siècles de travail en France*. INSEE, 1991.
L'évolution de la population active, et la montée accélérée du tertiaire depuis 1975.
- [202] Alain Desrosières et Laurent Thévenot. *Les catégories socio-professionnelles*. La Découverte, 2002.
- [203] Jean Tirole. *The Theory of Industrial Organization*. MIT Press, 1988.
Une exploration de la théorie de la concurrence imparfaite.
- [204] Alexis de Tocqueville. *De la démocratie en Amérique*, 1835.
- [205] Jean-Michel Truong. *Totalement inhumaine*. Les empêcheurs de penser en rond, 2002.
Illustration d'un fantôme.
Voir www.volle.com/lectures/truong.htm.
- [206] Alan M. Turing. Computing machinery and intelligence. *Mind*, (59), 1950.
L'article dans lequel Turing formule son fameux pari.
- [207] Pascal Roques et Franck Vallée. *UML en action*. Eyrolles, 2003.
Voir www.volle.com/lectures/roques.htm.
- [208] François-Xavier Verschave. *Noir silence*. Les Arènes, 2000.
Pour comprendre le mécanisme de la prédation.
Voir www.volle.com/lectures/verschave.htm.

- [209] Paul Virilio. *La bombe informatique*. Galilée, 1998.
- [210] Christophe Talière et Michel Volle. Modélisation technico-économique du réseau internet. Technical report, Eutelis, mars 1996.
www.volle.com/travaux/coutinternet.htm.
- [211] Michel Volle. *Le métier de statisticien*. Economica, deuxième édition, 1984.
- [212] Michel Volle. *Analyse des données*. Economica, quatrième édition, 1997.
- [213] Michel Volle. *e-économie*. Economica, 2000.
- [214] Michel Volle. Complexité et complication. *Génie logiciel*, (64), mars 2003.
- [215] Michel Volle. Fonctionnement d'un système informatique d'aide à la décision (siad). In *Revue des Nouvelles Technologies de l'Information, EGC 2004*, volume I. CEPAD, 2004.
- [216] Philippe Penny et Michel Volle. La téléinformatique dans l'entreprise. *La Recherche*, juin.
- [217] Bertrand Russell et Alfred Whitehead. *Principia Mathematica*, 1913.
- [218] Maurice Wilkes. Progress in computers. In *Prestige Lecture delivered to IEE*, Cambridge, février 2004.
www.cl.cam.ac.uk/~mvw1/Progress_in_Computers_IEE_Cambridge_2004_web.pdf.
- [219] Niklaus Wirth. *Algorithms + Data Structures = Programs*. Prentice-Hall, 1976.
- [220] G. H. Hardy et E. M. Wright. *An Introduction to the Theory of Numbers*. Oxford Science Publications, 1979.
Une bonne présentation de la théorie des nombres.
- [221] Frances A. Yates. *The Art of Memory*. Pimlico, 1966.
Voir www.volle.com/lectures/yates.htm.

Index

- Étude OFR, 432, 452, 460–462
Œuvre, 405
de Valence François, 14
3Com, 113, 118, 120
- Abelson Harold, 203, 232, 283, 581, 594
Abramson Norman, 341, 346, 355, 584
Acadys, 200
Access, 323
ACM (Association for Computing Machinery), 588, 589, 593
ACM (Association for Computing Machinery), 13, 333, 352
Adobe, 118
ADSL (Asymmetric Digital Subscriber Line), 9, 146, 354, 425
Adventure, 346
AFAI (Association Française des Auditeurs Informatiques), 15, 517
Aillagon Jean-Jacques, 167
Air France, 13–15, 152, 169, 399, 403
Air Inter, 14
Akers John, 118
Alcatel, 152
Aldus, 118
Algol, 282
All-in-One, 137
Allen Paul, 111, 112
Aloha, 341, 355–358, 593
Aloha discrétisé, 355, 357, 358
Alohanet, 341, 346
Altair, 51, 107, 109, 111, 112
AltaVista, 96
Altime, 14
Alto, 113, 114, 341
Amazon, 123
AMD, 51, 53, 121
Amdahl, 45, 126
American Airlines, 181, 190, 455, 480, 560
Amiga, 135
AMO (Assistant à maîtrise d’ouvrage), 405, 407, 409
Andrieux François, 14
Annuaire, 363, 371, 425
ANPE, 13–15, 99, 168, 228, 268, 274, 399, 400, 454
Aoki Masahiko, 50, 584
AOL, 119
AOS, 135
Apache, 121, 126, 130, 591
API (Application Programming Interface), 127
Apple, 51, 113–115, 117, 118, 120, 586
Appletalk, 133
Apple I, 113
Apple II, 25, 51, 113, 115, 135, 139
Apple III, 113
APT, 282
APU (Automate Programmable doué d’Ubiquité), 10, 30, 41, 200, 201, 203–205, 232, 444, 581
Arcome, 13, 14, 133
Areva, 152
Aristote, 234–237, 271, 317, 481
Arobase, 344
ARPA (Advanced Projects Research Agency), 330, 331, 333, 335, 341, 346, 347
Arpanet, 335, 339–341, 343, 344, 346–349
Ashton & Tate, 117
Ashton Alan, 134
AT&T, 49, 106, 173, 174, 322, 330, 332, 333, 335, 336, 346, 347, 577

- Atari, 135
- ATM (Asynchronous Transfer Mode), 339, 342, 425
- Augustin (Saint), 94
- Authentication, 276, 371, 399, 498
- AXA, 15
- Baan, 292
- Bachelard Gaston, 234, 584
- Backus John, 282
- Badillo Patrick, 14
- Banco Itamarati, 455
- Baran Paul, 173, 332, 333
- Bardeen John, 49
- Barnes & Noble, 123
- Basic, 111, 139, 282
- BBN, 333–335, 339, 346–348
- BEA, 426
- Beauvais Jean-Léon, 213, 584
- Bell Graham, 173
- Bell Labs, 49, 106, 282, 322, 348
- Bell System, 332
- Benci Guillaume, 292
- Bennahum David, 107, 346, 584
- Bergson Henri, 238
- Bernard Claude, 133
- Bernard Michel, 14
- Berners-Lee Tim, 118, 121, 349–351, 584
- Berst Jesse, 388
- Berthier Christophe, 14
- Beving Jean-Michel, 14
- Bezos Jeffrey, 123
- Billard Sylvie, 14
- BIPE, 132
- BITNET (Because It's Time Network), 348
- Blanc Christian, 14
- Blankenbaker John, 108
- Bloch Laurent, 13, 14, 24, 132, 283, 284, 454, 581, 584, 585
- Bloch Marc, 582
- Blondel Maurice, 42, 43, 210, 213, 224, 585
- Boileau-Despréaux Nicolas, 238
- Bon Michel, 577
- Booch Grady, 323, 324, 585
- Bordes-Pagès Gilles, 14
- Borland, 121, 140
- Bourbaki N., 33, 203, 204, 219, 224, 585
- Bourdieu Pierre, 150
- Bouveresse Jacques, 220, 222, 585
- Bouygues Telecom, 13, 14
- Bouzou Nicolas, 166
- Boydens Isabelle, 585
- Boydens Isabelle, 13, 81, 228, 271, 585
- BPML (Business Process Modeling Language), 77
- Braleret Denis, 14
- Brattain Walter, 49
- Breton Philippe, 104
- Bricklin Daniel, 139, 585
- Brown John, 183
- Brulé Jean-Pierre, 19, 49, 50, 504, 577, 585
- Buisson Jacques, 14
- Bull, 72, 103, 109
- Bus, 26, 109, 283, 295, 296, 337, 359–361, 495
- Busicom, 51
- Buzzati Dino, 484
- C, 106, 282, 283, 315, 322
- C++, 114, 128, 282, 283, 322, 323, 440, 552, 581, 594
- Cabouat Jérôme, 14
- Caillies, 72
- CALC, 139, 140
- California DMV, 454
- Camp L. J., 131, 585
- Campbell-Kelly Martin, 7, 289, 292, 585
- Carroll Paul, 115, 116, 335, 586
- Carruthers Mary, 215, 586
- Cary Frank, 115, 116
- Cas d'utilisation, 324, 325, 328, 415, 445, 446, 448–450
- Catenet, 346
- cc:Mail, 120
- CD-ROM, 119
- Census, 72
- Centre d'appel, 78, 135, 166, 177, 203, 274, 364, 375–379, 399–401, 427, 428, 433, 472, 490, 548, 568, 569
- Cerf Vint, 334, 340, 346–348, 589

- CERFnet, 348
 CERN (Conseil Européen pour la Recherche Nucléaire), 118, 121, 350
 Certicom, 120
 Ceruzzi Paul, 7, 104, 586
 Champy, 291
 Church-Turing (thèse de), 315
 CIGREF (Club Informatique des Grandes Entreprises Françaises), 15, 505
 Circuit virtuel, 29, 337
 Cisco, 113, 131, 335
 Clark James, 114
 Clark Wes, 333
 Clermont Markus, 141, 586
 Clot-Lafleur Ghislaine, 14
 Club des Maîtres d'Ouvrage des Systèmes d'Information, 13, 14, 505
 CNAM-TS, 505
 CNET (Centre National d'Études des Télécommunications), 13, 14, 132, 133
 CNIL (Commission nationale informatique et libertés), 186
 COBIT, 431
 Cobol, 282, 315, 323, 352, 552, 581
 Colas-Bara Matthieu, 14, 325
 Collet Laurent, 14
 Commodore, 115
 Commutation de paquets, 173, 332, 333, 336–338, 341, 343, 346–348
 Compaq, 117, 119, 124
 Compileur, 44
 Compuserve, 117, 119
 Computer Associates, 140, 426
 Confucius, 210, 231, 243, 586
 Contini Isabelle, 14
 Control Data, 333
 Coopers & Lybrand, 122
 Coordination des maîtrises d'ouvrage, 409, 410, 465
 Copernic, 505
 CORBA (Common Object Request Broker Architecture), 277, 295, 321
 Corel, 138
 Cottin Philippe, 14
 Coudreuse Jean-Pierre, 14, 339
 CP MOE (Chef de projet MOE), 411, 463
 CP/M, 135
 Crandall Robert, 190, 191, 560
 Crichton Michael, 153, 586
 Crockier Steve, 339
 Crown Books, 123
 Crowther Will, 346
 CSMA, 358
 CSMA-CD, 358, 359
 CSNET (Computer Science Research Network), 348
 CSSI (Comité Stratégique du Système d'Information), 397, 410, 423, 461, 481, 489
 Culture d'entreprise, 115, 218
 Curien Nicolas, 14
 CVS (Correction des variations saisonnières), 379, 380, 382, 383, 403, 558
 Cyclades, 346
 d'Aquin Thomas, 236–238
 Dahl Ole-Johan, 310, 322, 586
 Darbandi François, 14
 DARPA (Defense Advanced Projects Research Agency), 330
 DARPA (Defense Advanced Projects Research Agency), 330, 343, 344, 347, 348
 Data General, 48, 135
 Datagramme, 337, 340, 347
 Datamart, 308
 Datamining, 296–298, 300, 403, 426, 550
 Datawarehouse, 123, 297, 305–309, 403, 426, 550
 David Lionel, 14
 Davidoff Monte, 112
 Davies Donald, 173, 332, 333, 339, 346
 Davis Stan, 209, 591
 dBase, 117, 282
 DCA (Defense Communications Agency), 347
 DDE (Dynamic Data Exchange), 138

- de Mijolla Alain, 14
 de Pierrefeu Diane, 14
 de Saussure Ferdinand, 42, 46, 202, 248, 249, 251, 258, 593
 de Tocqueville Alexis, 66, 594
 Debaille Suzanne, 14
 Debonneuil Michèle, 14
 Debonneuil Xavier, 14
 DEC, 117, 121, 124, 135, 325, 335, 361
 Dell, 123, 124, 181
 Delsaux Michel, 14
 Department of Labor, 56
 Descartes René, 244, 353, 586
 Desfray Philippe, 14, 445
 Desrosières Alain, 14, 221, 594
 Diabolo, 135
 Diagramme d'état, 326
 Diagramme d'activité, 79, 326, 328, 329, 416, 445, 448
 Diagramme de cas d'utilisation, 326, 328
 Diagramme de classe, 326, 328
 Diagramme de collaboration, 326
 Diagramme de séquence, 326, 328
 Dierickx Bernard, 14
 Diffie Whitfield, 173, 369–371
 Digital, 39, 40, 92, 119
 Dijkstra Edsger, 86, 586
 Dirac Paul, 222
 Directeur de projet, 196, 409, 463–466, 520
 Displaywrite, 136
 DMCA (Digital Millenium Copyright Act), 131
 DNS (Domain Name System), 348
 Documentation électronique, 75, 78, 104, 132, 144, 193, 203, 253, 255, 342, 362–368, 376, 425, 426, 492, 553, 563
 Donjon & Dragon, 346
 Dostaler Gilles, 10, 162, 586
 Doweck Gilles, 203, 586
 DSI (Directeur du système d'information), 196, 295, 410, 411, 423, 429, 434, 465, 486–490, 497, 519, 570, 572–574
 DSI (Direction du système d'information), 82, 196, 293, 367, 406, 408–410, 434, 464, 487, 490, 507, 519, 520, 554, 567, 570, 571, 573
 du Castel François, 14
 Dubois Gérard, 14, 132
 Dupoty Jean-Claude, 14
 DVD, 119, 123, 575
 e-business, 97, 98, 185, 263, 402, 510, 548
 EAI (Enterprise Application Integration), 294–296, 321, 425, 428, 487, 491, 572
 eBay, 181
 Economica, 13
 EDI (Échange de données informatisé), 402
 EDS, 410
 EDSAC, 24
 EHO (Être Humain Organisé), 10, 200, 201, 203–205, 232, 444, 581
 EISA, 117
 Ekeland Ivar, 224, 586
 Ellis James, 173, 369–371
 Elmasri Ramez, 270, 274, 591
 ENIAC, 24, 25, 28, 29, 51, 123
 ENSPTT, 14
 Epson, 135
 Ericsson, 324
 ERP (Enterprise Resource Planning), 98, 289–294, 426, 487, 488, 491, 496, 518, 546, 572
 Escarpit Robert, 36, 586
 Estèbe Philippe, 14
 Ethernet, 20, 23, 113, 114, 117, 118, 120, 133, 337–339, 341, 342, 354, 359–361, 425
 Eutelis, 13–15, 133, 134, 595
 eWorld, 118
 Excel, 117, 139–141, 283, 323, 365, 389, 398
 Facci Hervé, 14
 Faggin Federico, 51
 FAI (Fournisseur d'accès à l'Internet), 342
 Fairchild Semiconductor, 50, 52

- Fakhfakh Abdelfatteh, 14
 Faure Jean-Marie, 13
 Favret-Saada Jeanne, 565, 587
 Fayon David, 14
 FBI, 197, 498, 499
 FDDI (Fiber Distributed Data Interface), 425
 FedEx, 122, 181, 480
 Felleisen Matthias, 587
 Felleisen Matthias, 132, 283, 587
 Feynman Richard, 587
 Feynman Richard, 225, 227, 543
 Fiévet Martine, 14
 Fielding Roy, 130, 591
 Figer Jean-Paul, 14, 87
 Flores Fernando, 134
 Focus, 282
 Forsythe George, 3, 587
 Fortran, 132, 133, 138, 139, 282, 315, 352, 552, 581
 Foucault Michel, 43, 221, 587
 Fournier Alain, 14
 Fowler Martin, 323, 439, 593
 Frame Relay, 339, 342, 425
 France Telecom, 13, 14, 56, 169, 173, 174, 188, 190, 404
 Frankston Bob, 139
 French American Foundation, 14
 Freud Sigmund, 241
 Friedman Daniel, 132, 587
 Frontline, 141
 FrontPage, 134, 284, 323, 352
 FTP (File Transfer Protocol), 343–345
 Fujitsu, 126
 Fylstra Daniel, 139

 Gödel (Théorème de), 84, 85, 203, 210, 219, 222, 468
 Gödel Kurt, 84, 85, 210, 230, 587
 Galilée, 5, 234, 238
 Garcin Michel, 14
 Gates Bill, 50, 96, 107, 111, 112, 125, 127, 128, 130, 143, 577, 587
 Gaulué Édouard, 14
 Gauthier Roger, 14
 Gem, 136
 Genelle Jacob, 14
 Gensollen Michel, 14, 131
 Gerstner Lou, 118, 497, 587
 Gille Bertrand, 3, 49, 164, 209
 Gleick James, 227, 230
 Gnumeric, 140
 Gordon Robert, 181
 Gourvennec Yann, 14, 181
 GPS (Global Positioning System), 144
 Gramsci Antonio, 66, 577
 Grandmont Jean, 208
 Gray Elisha, 173
 Grisham John, 153
 Groddeck Georg, 241
 Groupware, 133, 362, 375, 494
 groupware, 75, 118, 120, 122, 253, 426
 Grove Andy, 50, 124
 Guibert Bernard, 14, 207, 221, 239
 Guichard Jacques, 14
 Gurbaxani Vijay, 183

 Haberer Jean-Yves, 577
 Hafner Katie, 330, 335, 345, 348
 Hagel John, 183
 Hammer Michael, 291
 Handspring, 27
 Hardy G. H., 370
 Hatchuel Armand, 156
 Hegel Georg, 238
 Heidegger Martin, 150
 Hennessy John, 26
 Hennion Bernard, 14
 Henriët Dominique, 14
 Henry J. S., 133
 Herbsleb James, 130
 Hiltzik Michael, 113, 282, 335
 Hirji Karim, 296
 Hodges Andrew, 93
 Hoerni Jean, 50
 Hoff Ted, 51
 Honeywell, 111, 333, 335
 Hopper Grace, 44
 HP, 117, 118, 325, 335
 HTML (Hypertext Markup Language), 298, 349, 386
 HTTP (Hypertext Transfer Protocol), 298, 349
 Huin Jean-Pierre, 14
 Hume David, 209
 Husserl Edmund, 210

- Hypercube, 229, 300, 301, 304, 308, 309
- IBM, 25, 29, 45, 51, 71, 73, 97, 105, 113–121, 123–127, 135–137, 139, 140, 173, 174, 282, 325, 333, 335, 336, 348, 361, 410, 425, 426, 497, 577
- IBM Global Services, 127
- Idéliance, 15
- IEEE, 13
- iMac, 120
- IMP (Interface Message Processor), 333–335, 341, 343, 347, 348
- Informatique de communication, 362
- Informix, 425
- Ingals Dan, 114
- INRA, 108
- INSEE, 13, 55
- Intel, 26, 49–53, 55, 103, 105, 108, 109, 115–119, 121, 124, 361, 367
- Internet, 9, 20, 52, 59, 76, 78, 98, 104, 106, 111, 113, 119–121, 123, 125, 128, 129, 131, 173, 174, 262, 284, 298, 331, 332, 337, 338, 342, 346–349, 371, 375, 495
- Internetting Project, 346
- Intranet, 20, 75, 76, 104, 119, 120, 133, 144, 253, 263, 296, 298, 299, 328, 342, 362, 365, 366, 368, 375, 383, 386, 388, 399, 431, 433, 436, 494, 495
- Intriligator Michael, 537, 539
- INWG (International Network Working Group), 346
- IP (Internet Protocol), 339, 340
- IPTO (Information Processing Techniques Office), 331
- ISO (International Standard Organization), 42, 340
- ITIL, 431
- J2EE, 321
- Jacobson Ivar, 323, 324
- Jacq Francis, 14
- Jancovici Jean-Marc, 14, 27
- Java, 114, 129, 178, 277, 282, 440, 581
- JD Edwards, 292
- Jean Gérard, 14, 433
- Jobs Steven, 113
- Joskowicz Jean, 14
- Jouannin Yannick, 14
- Joule Robert-Vincent, 213
- Jullien François, 168, 213, 225
- Kahn Bob, 340, 346
- Kahn Philippe, 108
- Kalevala, 128
- Kandel Eric, 215, 229
- Kannounikoff Dimitri, 14
- Kant Emmanuel, 209, 239
- Kapor Mitch, 140
- Kasparian Jean-Jacques, 14
- Kay Alan, 53, 114, 128, 282, 322
- Keen Peter, 373, 532
- Kenbak-1, 108
- Kent William, 272
- Keynes John Maynard, 10, 208
- Khiplea, 14, 325
- Kidder Tracy, 143
- Kierkegaard Søren, 238
- Kilby Jack, 50
- Kleinrock Leonard, 334
- Knuth Donald, 3, 6, 33, 44, 45, 71, 83, 86, 103, 201, 283, 289, 354, 370, 581
- Kodak, 118
- Korzybski Alfred, 223
- Kott Jean, 13
- Kroc Ray, 544
- KSpread, 140
- Lévy Michel-Louis, 14
- Lévy Pierre, 87, 209
- Laganier Jean, 14, 207, 221, 239
- Lai Vincent, 296, 298
- Lampson Butler, 114
- LAN (Local Area Network), 337, 354
- Landau Lev, 224
- Landweber Larry, 348
- Langage, 31
- Langage conceptuel, 33
- Langage connoté, 34
- Laudauer Thomas, 181

- Laurès Antoine, 14, 133
 Le Diberder Alain, 14, 132, 344
 Le Guyader Hervé, 43, 221
 Le Moigne Jean-Louis, 209
 Leblond Thierry, 14
 Lecointre Guillaume, 43, 221
 Lefebvre René, 14
 Legendre Pierre, 209
 Leibniz Gottfried, 173
 Leibowitz Yeshayahou, 226
 Lejemble Mallory, 14
 Lereau Hervé, 14
 Lerner Josh, 128, 182
 Levasseur Lionel, 14
 Levy Steven, 31, 107, 369
 Lickliger Joseph, 331, 333, 349
 Lifchitz E., 224
 Ligonnière Robert, 104
 Linux, 121, 126, 128, 140, 425, 493
 Lisa, 51, 113, 117, 120
 LISP, 132, 282, 581
 Litwin Witold, 14
 Logiciel Libre, 106
 Logiciel libre, 106, 120, 125, 128
 Longépé Christophe, 14
 Lotus, 117, 119, 120, 136, 140, 142
 Lotus 1-2-3, 117, 137, 140, 141
 Lotus Domino, 426
 Lotus Notes, 118, 120, 133, 299, 375
 Louis XIV, 194
 Lukasik Stephen, 344
 Luttwak Edward, 577
 Lyon Matthew, 330

 Médiamétrie, 27
 Mèlèse Jacques, 198
 Maître d'œuvre, 405
 Maître d'ouvrage, 405
 Maîtrise d'ouvrage, 7
 Maîtrise d'ouvrage (n'est pas un concept
 franco-français), 406
 Macintosh, 51, 113, 114, 120, 133,
 135, 140, 141, 144
 MacKenzie Kevin, 345
 MacOS, 120
 Malinvaud Edmond, 14
 Mandel René, 14
 Mangonaux Michel, 14
 Marc Edmond, 257

 Marchand Marie, 14
 Marchand Olivier, 158
 Mark-8, 109
 Marlowe Jos, 178, 282
 Marrill Tom, 333, 339
 Mattessich Richard, 138
 McCarthy John, 282
 McDonald Maurice, 544
 McDonald Richard, 544
 MCI, 117
 McKinsey, 152
 McLuhan Marshall, 41
 McNealy Scott, 178, 282
 MDA (Model Driven Architecture),
 321
 MEDEF, 195, 576
 Meinadier Jean-Pierre, 255, 258
 Menciùs, 210
 Merise, 79, 324, 431
 Messagerie, 342, 345, 387, 551, 553
 Metcalfe Robert, 114, 341
 Meyer Christopher, 209
 MHS (Message Handling Service), 120
 Micral, 51
 Micropro, 135, 136
 Microsoft, 111, 113, 115–122, 124, 126–
 128, 130, 135–138, 140, 141,
 143, 174, 284, 325, 367, 375,
 425
 Microsoft Exchange, 375, 426
 Microsoft Works, 140
 Middleware, 283, 321, 493
 Minc Alain, 202
 MIPS, 51
 MIT, 139, 282, 335
 MITS, 109, 111, 112
 Mitterrand François, 454
 MMU (Memory Management Unit),
 286
 MOA (Maîtrise d'ouvrage), 405
 MOAD (Maître d'ouvrage délégué),
 407
 MOAO (Maître d'ouvrage opération-
 nel), 197, 407, 408
 MOAS (Maître d'ouvrage stratégique),
 407
 Mobil Oil, 181
 Mockus Audris, 130

- Modèle en couches, 339
 Modem, 336, 342
 MOE (Maîtrise d'œuvre), 405, 420
 Monde de la nature, 212
 Monde de la pensée, 212
 Monde des valeurs, 215, 216
 Montefiore Simon, 244
 Moore (Loi de), 50, 52, 496
 Moore Gordon, 50, 52, 53
 Morand Bernard, 79
 Morel Michel, 456
 Morin Edgar, 209
 MOS Technologies, 51
 Mosaic, 121, 140, 349, 350
 MOST, 15
 Motorola, 51, 53, 118, 121
 Mott Tim, 114
 MP3, 52
 MQSeries, 426
 MS-DOS, 135, 137, 138, 285
 MSA, 14
 MSG, 345
 MsgGroup, 345
 MTP (Message Transfer Protocol), 345
 Multiplan, 117, 132, 133, 140
 Musso Pierre, 14

 Nadeau Tom, 130
 Nadel Henri, 14
 NASA, 330, 348
 Navathe Shamkant, 270
 NCP (Network Control Protocol), 339, 340, 346
 Needham Joseph, 8, 234
 Netscape, 118, 119, 350
 Netscape Enterprise, 126
 NetWare, 120
 Network Measurement Center, 334
 Newman Max, 24
 Newton Isaac, 173
 Nomia, 279, 329, 426
 Nora Simon, 202
 Notebook, 118
 Novell, 118, 120, 122, 124, 138
 Noviant Jacky, 14
 Noyce Robert, 50
 NSA (National Security Agency), 371
 NSF (National Science Foundation), 347, 348
 NSFNET, 348, 349
 Numérique, 39
 Numéris, 425
 NWG (Network Working Group), 339
 Nygaard Kristan, 322
 Nygaard Kristen, 310
 NYSERNET, 348

 O'Beirne Patrick, 141
 Objet, 35
 OEM (Original Equipment Manufacturer), 122
 OMG (Object Management Group), 321
 OnMap, 279, 329, 426
 OpenOffice, 138
 Oracle, 292, 325, 425
 Oracle Applications, 292
 ORB (Object Request Broker), 321
 OS/2, 117, 137, 138
 OSI (Open Systems Interconnection), 42, 335, 340, 341
 Ott Didier, 14
 Outsourcing, 118, 119
 Ouvrage, 405

 PABX (Private Automatic Branch Exchange), 373, 398
 Palmtop, 26, 399
 Palo Alto (École de), 485, 557, 592
 Panko Ray, 141
 Paperback, 140
 PARC (Palo Alto Research Center), 106, 113, 114, 117, 282, 322, 341, 346
 Parménide, 5, 235
 Pascal, 282, 283
 Pascal Blaise, 226, 238
 Passerelle, 338, 340, 347
 Patterson David, 26
 Pattis Richard, 31
 Paulson Linda, 130
 Pavé Francis, 41
 Pavlevski Jean, 13
 PC, 115, 116
 Peaucelle Jean-Louis, 14, 531
 Penny Philippe, 14, 133

- Pentium, 51, 118, 119, 121
- Pentium Pro, 119
- Peoplesoft, 292
- Pepsi-Cola, 139
- Perl, 128, 129, 282, 581
- Perret Jacques, 30
- Personnalité de l'entreprise, 218
- PET, 115
- Peterson Pete, 134
- PGP (Pretty Good Privacy), 119, 371
- Picard Dominique, 257
- Pilhes René-Victor, 153
- Platon, 93, 235
- Plazanet Philippe, 14
- Ploquin Lionel, 14
- Plotin, 94
- Poincaré Raymond, 239
- Pont, 337
- POP (Post Office Protocol), 345
- Popper Karl, 42, 89, 93, 224, 227
- Portefeuille applicatif, 412
- Portefeuille de projets, 412
- Positionnement, 218
- Postel John, 345
- Pouzin Louis, 339, 346
- Power AMC, 426
- Powerbook, 118
- PowerBuilder, 426
- PowerPC, 51, 120, 121
- PowerPoint, 323, 387, 389
- Powers James, 72
- Presentation Manager, 137
- Price Waterhouse, 122
- Pringault Alex, 14
- Printz Jacques, 14, 461, 522, 581
- Prodigy, 119
- Progress AppModeller, 426
- Prolog, 581
- Protocole, 333, 336, 340
- Proust Marcel, 40, 353
- PS/2, 117, 118, 121
- Python, 581

- Quérou Yvon, 14
- Quattro, 140
- Queuille Henri, 100
- Quire Larry, 215

- R/2, 291
- R/3, 291
- R2E, 51, 108
- Réchaussat Louis, 15
- Référentiel, 495
- RAD (Rapid Application Development), 121
- RAM (Random Access Memory), 286
- Rambourdin Michel, 15
- RAND, 335
- Rational, 325
- Rational Rose, 283, 433
- Rawls John, 224
- Raymond Eric, 129, 284
- RCA, 126
- Renaud Olivier, 15
- Reuters, 181
- RFC (Request for Comments), 339
- RISC, 120
- RISC (Reduced Instruction Set Computer), 26
- Rivière Pascal, 13
- RLPC (Réseau Local de PC), 354
- RLPC (Réseau local de PC), 337
- Roberts Ed, 109
- Roberts L. G., 357
- Roberts Larry, 331, 333
- Rochet Claude, 15
- Rock Art, 50
- Rockefeller John D., 174
- Rohmer Jean, 15
- Romer Paul, 176
- Roques Pascal, 323, 449
- Rosen Benjamin, 139
- Rossi Serge, 104
- Rousseau Raphaël, 14
- Routeur, 337
- RPV (Réseau Privé Virtuel), 342
- RTC (Réseau téléphonique commuté), 425
- Rumbaugh James, 323, 324
- Russell Bertrand, 85, 230

- Saint-Simon Louis de, 191, 194
- Sammet Jean, 281
- SAP, 14, 290-292, 308
- SAS, 298, 308, 426
- Satnet, 346, 347
- Sauer Jean-François, 15
- Scelbi, 109

- Scheme, 132, 282, 283
- Schneier Bruce, 95
- Schwartz Laurent, 203
- Scott Kendall, 323
- Sears et Roebuck, 123
- Semi-conducteur, 49
- Sendmail, 130
- Serlet Jean-Claude, 14
- Serres Michel, 87
- Sfez Lucien, 206
- SGBD, 292
- SGBD (Système de gestion des bases de données), 425
- Shannon Claude, 36
- Shockley Robert, 49
- SIAD (Système informatique d'aide à la décision), 426
- SIAD (Système informatique d'aide à la décision), 299, 550
- Silicium, 47
- Silicon Graphics, 51, 114
- Simon Herbert A., 198
- Simonyi Charles, 114
- Simula, 282
- Singh Simon, 370
- Single Sign On, 399
- Sipser Michael, 84, 86
- Sjobbema DJW, 104
- Smalltalk, 114, 128, 282, 322
- Smith Adam, 582
- SMTP (Simple Message Transfer Protocol), 345
- SNCF, 454
- SOA (Services Oriented Architecture), 204, 277
- Softeam, 15, 445
- Software Arts, 139, 140
- Solaris, 121
- Solow (Paradoxe de), 117
- Solow (paradoxe de), 181, 504
- Solow Robert, 117, 176, 181, 504
- SPAN (Space Plasma Analysis Network), 348
- SPARC, 51, 121
- Sperry Rand, 72
- Spinetta Jean-Cyril, 15
- Spoutnik, 330
- SQL, 282
- SQL Server, 425
- Squire Larry, 229
- SRI (Stanford Research Institute), 334
- SSEC, 123
- SSI, 135, 136
- Standish Group, 197, 453, 454, 517
- Star, 114
- Starkweather Gary, 114
- Stiller Richard, 178, 282
- Strassmann Paul, 183, 487, 572
- Stroustrup Bjarne, 128, 282, 322
- Sun, 51, 282, 321, 341
- SuperCalc, 140
- Sussman Gerald, 203, 232, 283, 581
- Sutherland Ivan, 331
- Sybase, 426
- Syme S., 131
- System R, 290
- System Software, 292
- Télécommunications, 493
- Tableur, 138
- Tabourier Yves, 199
- Tache aveugle, 218, 564
- Talière Christophe, 15
- Tandy, 135
- Tanenbaum Andrew, 42, 45, 284
- Taylor Robert, 331, 346
- TCP (Transmission Control Protocol), 340, 343, 347
- TCP/IP, 144, 298, 322, 335, 337-342, 347-349, 425
- Teisserenc Patrick, 15
- Telenet, 348
- Telnet, 339, 343
- Telser Larry, 114
- Teradata, 123
- Texas Instruments, 51
- Textor, 133
- Thélot Claude, 158
- Thévenot Laurent, 221
- Thacker Chuck, 114
- ThinkPad, 118
- Thonnet Michèle, 15
- TIP (Terminal IMP), 335, 343
- Tirole Jean, 128, 182
- Titus Jonathan, 109
- TMA (Tierce Maintenance Applicative), 411

- Toile, 15, 29, 98, 104, 119–122, 134, 144–146, 150, 173, 193, 278, 298, 321, 323, 349–353, 362, 364, 388, 402, 428, 433, 436, 448, 498, 499, 548, 584
- Token Bus, 360
- Token Ring, 337, 354, 360
- Tomlinson Ray, 344
- TopView, 136
- Torvalds Linus, 128, 130
- Transistor, 49
- Treo, 27
- Truffier Claude, 15
- Truong André, 108
- Truong Jean-Michel, 87, 202
- Turbo Pascal, 121
- Turing (machine de), 86, 315
- Turing (pari de), 90, 94
- Turing Alan, 86, 89, 90, 92–94, 202, 315, 369
- Tuxedo, 426
- UCLA, 334, 339
- UCSB, 334
- UML (Unified Modeling Language), 551
- UML (Unified Modeling Language), 77, 79, 80, 204, 279, 310, 314, 320, 321, 323–326, 328, 329, 414, 434, 439, 445, 448
- Unbundling, 126
- Unisys, 325
- United Airline, 122
- Univac, 124
- Unix, 117, 122, 135, 282, 341
- URI (Uniform Resource Identifier), 351
- URL (Uniform Resource Locator), 349
- US Postal Service, 345
- UUCP (Unix-to-Unix Copy Program), 348
- Vallée Franck, 323, 449
- Varin Julien, 15
- VAX, 135, 137
- Veille SI, 496
- Verschave François-Xavier, 6, 582
- VGA (Video Graphics Array), 121
- Victor, 135
- Vidal Paul, 15
- Virilio Paul, 39, 87
- VisiCalc, 139, 140
- Visicalc, 113
- VisiCorp, 139, 140
- Visual Basic, 121
- Vittal John, 345
- von Neumann (architecture de), 72, 123, 284
- von Neumann John, 24, 72, 88, 203
- W3C (World Wide Web Consortium), 350, 351
- Wal-Mart, 123, 181
- Wall Larry, 128, 282
- WAN (« Wide Area Network »), 118
- WAN (Wide Area Network), 339, 425
- Wang, 124, 134, 136
- Watson Thomas J., 71
- Wearable Computer, 27, 145
- Web, 494
- Web Services, 277, 493
- Whitehead Alfred, 85, 230
- WiFi, 342
- Wilder Billy, 28
- Wilkes Maurice, 24, 26
- Windows, 120, 122, 133, 136–138, 140, 144, 285, 425
- Windows 1.0, 117
- Windows 95, 119
- Windows NT, 118, 119
- Winograd Terry, 134
- Wirth Niklaus, 128
- Wittgenstein Ludwig, 151
- Woods Don, 346
- Word, 135, 138, 283, 387
- WordPerfect, 118, 135–138
- Wordstar, 135, 136
- Workflow, 76–78, 134, 185, 203, 255, 311, 313, 314, 364, 366–369, 373–376, 389, 404, 425, 433, 438, 439, 494, 495, 551, 563, 571
- World Wide Web, 121
- Wozniak Steven, 113
- Wright E. M., 370
- Wysiwyg, 137
- X25, 337, 339, 342, 425

- Xénophon, 93
- Xerox, 106, 113–115, 120, 165, 173,
282, 322, 335, 341, 361, 552
- XML (eXtensible Markup Language),
122, 299, 349

- Yablonsky Serge, 15
- Yates Frances, 215, 241

- ZDNet, 388
- Zeitoun Jean, 15
- Zimmermann Philip, 119, 371

Table des matières

Introduction	3
Remerciements	13
I Comment l'Automate Programmable doué d'Ubiquité assiste l'Être Humain Organisé	17
1 Du côté de l'ordinateur	19
1.1 Un changement du rapport avec la nature	21
1.2 Qu'est-ce qu'un ordinateur?	23
1.3 Qualité du vocabulaire	28
1.3.1 « Ordinateur »	29
1.3.2 « Langage »	31
1.3.3 « Objet »	35
1.3.4 « Donnée » et « Information »	36
1.3.5 « Numérique »	39
1.4 Modèle en couches	42
1.4.1 Couches de l'ordinateur	43
1.4.2 Portée du modèle en couches	46
1.5 Vers l'ordinateur individuel	47
1.5.1 Origines et évolution du microprocesseur	50
1.5.2 La loi de Moore	52
1.5.3 Évolution du prix des micro-ordinateurs	55
1.6 Conjoncture des TIC	56
2 Automatisation et intelligence	66
2.1 L'informatisation de l'entreprise	67
2.1.1 Organisation du travail de bureau: années 1880	68
2.1.2 Arrivée de l'informatique: années 1950	71
2.1.3 La bureautique communicante: années 1980	74
2.1.4 Du concept au processus: années 1990	76

2.2	Entropie du système d'information	79
2.3	Limites de l'informatique	83
2.4	Quelle intelligence?	86
2.4.1	Turing a-t-il perdu son pari?	90
2.4.2	Articuler l'ordinateur et l'être humain	97
3	Éclairage historique	103
3.1	Histoire du micro-ordinateur	105
3.1.1	Les apports du PARC de Xerox	113
3.1.2	IBM et le PC	115
3.1.3	Chronologie du PC	117
3.2	Points de repère	119
3.2.1	Innovations importantes	119
3.2.2	Réussites exemplaires	122
3.3	Le marché de l'informatique	123
3.3.1	Dynamique des équilibres	125
3.3.2	Naissance du marché du logiciel compilé	126
3.3.3	Le modèle du logiciel libre	128
3.3.4	Pirates et protection	130
3.4	Les logiciels de bureautique	132
3.4.1	Histoire du traitement de texte	134
3.4.2	Histoire du tableur	138
3.4.3	Leçons de l'histoire de la bureautique	143
3.5	Esquisse de prospective	144
II	Le côté de l'entreprise	148
4	Qu'est-ce qu'une <i>entreprise</i>?	150
4.1	Le point de vue du système d'information	152
4.2	Que produit-on?	157
4.2.1	À propos de l'immatériel	160
4.2.2	Pour une économie de la qualité	162
4.3	Mission et organisation	167
4.4	Le moteur de l'innovation	172
4.5	Le commerce de la considération	177
5	À la recherche de la stratégie	179
5.1	L'informatique est-elle stratégique?	181
5.2	L'alignement stratégique	184
5.3	Du côté des dirigeants	187
5.4	La stratégie retrouvée	190
5.5	Restaurer le mot « Informatique »	198

6	Aspects philosophiques	206
6.1	Articuler la pensée et l'action	210
6.2	Complexité et complication	219
6.3	Apports de l'informatique à la philosophie	231
6.3.1	Un terrain d'expérimentation	233
6.3.2	À l'origine de la pensée occidentale: l'Être	234
6.3.3	Opacité de l'existant	236
6.3.4	De la pensée à l'action	239
6.3.5	De l'action aux valeurs	242
III	L'informatisation de l'entreprise	246
7	Socle sémantique	248
7.1	Langage et « langage »	248
7.2	Ingénierie du système d'information	252
7.3	Approche linguistique	258
7.3.1	Physique de l'entreprise	260
7.3.2	Articulation entre physique et sociologie	262
7.4	Administration des données	265
7.5	Bases de données	270
7.6	Lisibilité du système d'information	274
8	Outils et architecture	281
8.1	Langages de programmation	281
8.2	Systèmes d'exploitation	283
8.3	Découpage en « applications »	286
8.3.1	Les ERP	289
8.3.2	Les EAI	294
8.4	Intranet et Datamining	296
8.4.1	Datamining	296
8.4.2	Intranet	298
8.5	Le système informatique d'aide à la décision	299
8.5.1	Étude de cas: histoire d'un Datawarehouse	305
8.5.2	Annexe: Les hypercubes	309
9	Modélisation par objets	310
9.1	Des processus aux objets	310
9.2	De la programmation impérative à l'objet	314
9.2.1	Programmation impérative	315
9.2.2	Programmation par objets	317
9.2.3	Origines de la programmation par objets	322
9.3	Langage de modélisation UML	323

9.3.1	Histoire des modélisations par objets	324
9.3.2	UML en œuvre	325
9.3.3	Comment présenter un modèle UML?	328
10	La conquête de l'ubiquité	330
10.1	Naissance du réseau	330
10.1.1	La commutation de paquets	332
10.1.2	Le premier réseau d'ordinateurs	333
10.1.3	Problèmes de mise en place	335
10.1.4	Protocoles et paradigmes	336
10.1.5	TCP/IP et Ethernet	339
10.1.6	Premières applications	343
10.1.7	Vers l'Internet	346
10.2	Naissance de la Toile	349
10.3	Lire et publier sur la Toile	350
10.4	Annexe: Réseaux et protocoles	354
10.4.1	Largeur de bande, débit utile et « overhead »	354
10.4.2	Aloha	355
10.4.3	Carrier Sense Multiple Access	357
10.4.4	Ethernet	359
10.4.5	Protocoles sans collision	359
10.4.6	Domaines d'utilisation	361
11	L'informatique de communication	362
11.1	Les outils de communication	362
11.2	Place dans le système d'information	366
11.3	Sécurité	369
11.3.1	Système à clé publique	369
11.3.2	Identification, authentification et habilitation	371
11.4	Les workflows	373
11.5	Études de cas	375
11.5.1	Infotel	375
11.5.2	Histoire d'un tableau de bord	379
11.6	Annexe: Bon usage de la messagerie	387
IV	L'informatique dans l'organisation	392
12	Acteurs, fonctions et rôles	394
12.1	Le marketing interne	394
12.1.1	Besoins généraux	398
12.1.2	Besoins propres à la fonction dans l'entreprise	399
12.1.3	Une segmentation des données	404

12.2	Organisation et responsabilités	405
12.2.1	Fonctions dans la maîtrise d'ouvrage	406
12.2.2	Fonctions dans la maîtrise d'œuvre	410
12.2.3	Rôle de la maîtrise d'ouvrage	412
12.2.4	Rôle de la maîtrise d'œuvre	420
12.3	Articuler l'expertise et la décision	421
12.4	Compétences de la maîtrise d'ouvrage	423
12.5	Défis de l'informatique aujourd'hui	427
13	Méthodes	430
13.1	Urbaniser un système d'information	433
13.2	Approche par les processus	436
13.3	Modélisation	440
13.3.1	« Modéliser » l'entreprise	442
13.3.2	Démarche de modélisation	445
13.3.3	Quelle est l'utilité du modèle métier?	450
13.3.4	Modélisation et évaluation du coût	452
13.4	Réalisation des projets	453
13.4.1	La première expression de besoin	456
13.4.2	Étude Opportunité, Faisabilité, Risques	461
13.4.3	Fiche de synthèse de mission	463
13.4.4	Conduite de projet	463
13.4.5	Recette	468
13.4.6	Déploiement	470
13.4.7	Évaluation	472
14	Vers la maturité	474
14.1	Professionnalisation de la MOA	477
14.2	Servitudes et grandeur	480
14.2.1	Le côté de la maîtrise d'ouvrage	480
14.2.2	Le côté du DSI	486
14.3	Check-list du système d'information	491
14.4	Un système d'information mûr	494
14.5	Études de cas	497
14.5.1	Le système d'information d'IBM	497
14.5.2	Le système d'information du FBI	498
V	L'informatique dans le fonctionnement	500
15	Économie de l'informatique	502
15.1	Enjeu macroéconomique	503
15.2	Un facteur de production	506

15.2.1	Évaluer une entreprise	511
15.2.2	Économie et urbanisation	516
15.2.3	Budget de l'informatique	518
15.3	Calcul économique	521
15.3.1	Rentabilité d'un ensemble de projets	522
15.3.2	Durée de vie d'un logiciel	526
15.3.3	Coût des PC en réseau	529
15.3.4	Règles de gestion et optimisation	531
15.3.5	À la recherche du taux d'informatisation optimal . . .	537
16	Pathologie de l'entreprise	543
16.1	Sociologie de l'entreprise	543
16.1.1	Style de la direction	545
16.1.2	Un reflet des valeurs	547
16.1.3	Pôles de légitimité	548
16.2	Crise de l'entreprise	554
16.3	Obstacles au changement	557
16.3.1	La tache aveugle	564
16.3.2	Le compromis managérial	566
16.3.3	Le centre d'appel : un révélateur	568
16.3.4	Corruption et honnêteté	569
16.3.5	La tentation du DSI	572
16.4	La demi-stratégie	575
	Conclusion	580