

Une micro-informatique conviviale et intelligente

Gérard Mercure

Volume 35, numéro 3, juillet–septembre 1989

URI : <https://id.erudit.org/iderudit/1028151ar>

DOI : <https://doi.org/10.7202/1028151ar>

[Aller au sommaire du numéro](#)

Éditeur(s)

Association pour l'avancement des sciences et des techniques de la documentation (ASTED)

ISSN

0315-2340 (imprimé)

2291-8949 (numérique)

[Découvrir la revue](#)

Citer cet article

Mercure, G. (1989). Une micro-informatique conviviale et intelligente. *Documentation et bibliothèques*, 35(3), 113–119.
<https://doi.org/10.7202/1028151ar>

Tous droits réservés © Association pour l'avancement des sciences et des techniques de la documentation (ASTED), 1989

Cet document est protégé par la loi sur le droit d'auteur. L'utilisation des services d'Érudit (y compris la reproduction) est assujettie à sa politique d'utilisation que vous pouvez consulter en ligne.

<https://apropos.erudit.org/fr/usagers/politique-dutilisation/>

érudit

Cet article est diffusé et préservé par Érudit.

Érudit est un consortium interuniversitaire sans but lucratif composé de l'Université de Montréal, l'Université Laval et l'Université du Québec à Montréal. Il a pour mission la promotion et la valorisation de la recherche.

<https://www.erudit.org/fr/>

en périphérie

Une micro-informatique conviviale et intelligente

Deux mondes parallèles

La micro-informatique générale et la micro-informatique documentaire sont deux mondes voisins, mais qui se parlent peu. La micro-informatique documentaire préfère, pour ses bases de données bibliographiques, une structure d'organisation fondée sur la création de fichiers inversés et l'utilisation du champ à longueur variable pour y loger des données textuelles. La micro-informatique générale favorise le modèle relationnel qui convient mieux au traitement des données numériques du monde des affaires. Peu de bases de données bibliographiques ou de bibliothèques empruntent ce modèle sinon celles qui sont réalisées avec des Systèmes de gestion de bases de données (SGBD) relationnels... Un incident illustre la distance qui sépare ces deux mondes. En 1988, *PC Magazine*¹ faisait l'analyse comparative de 38 SGBD programmables pour PC et passait sous silence le fait que Advanced Revelation, l'un de ce groupe, était le seul qui offrait la possibilité de champs à longueur variable. Il fallut le rappel de la part d'un lecteur dans un courrier suivant de la revue pour signaler cette omission.

Depuis quelque temps cependant, ce monde de l'entreprise découvre les vertus de la base de données textuelle avec ces gestionnaires d'information personnelle, dits « information manager » tels que GrandView et Agenda. Ces utilitaires de bureau sont destinés au gestionnaire pressé. Ces bloc-notes électroniques permettent d'y consigner rendez-vous, notes de service et correspondance d'affaires, sans pour autant devoir accorder plus de temps qu'il ne faut au classement.

D'autres logiciels, plus structurés, tels que askSam, Tracker permettent de construire de véritables bases de données textuelles avec un minimum d'analyse préalable. Les bibliothécaires sont même devancés sur leur propre terrain par

des innovations dont ils auraient dû être, en toute vraisemblance, les inventeurs, tel l'hypertexte, par exemple... Qu'est-ce que l'hypertexte, en effet, sinon le livre non linéaire ou la bibliothèque dans laquelle on butine d'un concept à un autre et bouquine d'un média à un autre.

Bien qu'il y ait rapprochement, ces deux mondes ne partagent pas les mêmes perspectives ni ne rencontrent les mêmes contraintes. L'informatique grand public est plus intéressée à la recherche « dans le texte » qu'au signalement bibliographique comme s'y applique l'informatique documentaire. Pour réaliser une base de données, le bibliothécaire se préoccupe de la capacité de stockage et de la diversité des clés d'accès à la base. Les éditeurs de logiciels généraux recherchent la simplicité d'utilisation et la rapidité d'exécution. Le monde de la documentation est davantage gêné par la limite de capacité de stockage d'un disque logique à 32 Mo que par la limite de la taille d'un programme à 640 Ko. On observe toutefois que presque tous les logiciels documentaires s'approchent eux aussi maintenant de cette capacité limite de mémoire vive pour fonctionner. Les logiciels généraux, de leur côté, sont déjà à l'étroit. Ainsi Lotus 1-2-3 paraîtra désormais en deux versions, la v. 2.2 pour les appareils bas de gamme et la v. 3.0 pour des appareils plus puissants. C'est que les logiciels prennent de l'ampleur et enflent de version en version comme la grenouille de la fable.

Ezra Shapiro, chroniqueur au *PC Magazine*², réagit contre cette tendance inflationniste de la micro-informatique. Les utilitaires accessoires et les fonctions secondaires, selon lui, finissent par masquer la vraie raison d'être du logiciel d'origine. Avec la multiplication des fonctions et des menus, les logiciels actuels n'offrent plus cette élégance des premières versions. Les nouvelles versions superpuissantes n'ont plus cette concision, ce « sense of rightness » qui fait qu'on préfère

1. « Programmable Databases : dBase and its Challengers », *PC Magazine* (May 17, 1988), 93-272.

2. Ezra Shapiro, « Is bigger better? », *Byte*, vol. 14, no. 7 (July 1989), 125-128.

d'emblée un logiciel à un autre. Il invite l'industrie à un retour aux fonctions de base, simples et conviviales.

Mais, il y a une autre raison. Comme l'explique Philippe Kahn, le directeur de Borland :

L'augmentation du volume de code ne vient pas du coeur de l'application proprement dite. Aujourd'hui, deux choses sont réellement consommatrices de mémoire et de calcul : la gestion du graphisme et l'interface utilisateur³.

Pour offrir l'interface graphique, les développeurs se voient dans l'obligation de franchir la barrière du 640 Ko de mémoire vive et de passer au système d'exploitation OS/2 et à son interface graphique Presentation Manager. Or ces derniers ne fonctionnent décemment que sur un micro-ordinateur rapide (25 Mhz) doté de 4 Mo de mémoire vive, soit sur des machines de deux à trois fois plus rapides et ayant sept fois plus de mémoire vive (ou quatre Mo de mémoire) que les ordinateurs personnels les plus répandus (PC, XT et PS 30).

Les interfaces graphiques

Les interfaces graphiques n'ont pas pour seul but d'offrir un plus grand confort visuel et une meilleure ergonomie que les interfaces textuelles. À leur plus grande facilité d'utilisation se joint leur capacité d'uniformiser les commandes d'une application à l'autre. C'est pourquoi on les appelle parfois des « intégrateurs graphiques ». En plus de présenter les différentes applications sous un même décor, elles facilitent l'installation de chacune d'elles et des périphériques qui s'y rattachent. Car c'est l'interface graphique qui assure la liaison avec ce matériel plutôt que chacun des programmes. Elles permettent en outre de démarrer plusieurs programmes en même temps et, sur les ordinateurs plus puissants, de traiter plusieurs tâches concurremment et d'échanger l'information de l'une à l'autre.

L'interface graphique comprend généralement une souris, des menus affichables contrôlés par la souris, des fenêtres annonçant ce que l'ordinateur fera, soit par des icônes ou le nom du programme donnant accès aux répertoires, fichiers et programmes et enfin des boîtes de dialogue, des boutons, règles coulissantes et autres symboles graphiques.

Chaque interface se caractérise par sa ligne et sa tenue de route, « son look and feel », mais en

fouillant au-delà des caractéristiques particulières de chacune, on retrouve des traits communs.

Le numéro de juillet 1989 de *Byte* dresse la carte de ce nouveau continent des interfaces graphiques⁴. En 1984, il n'était occupé que par Apple et son Macintosh. Aujourd'hui, en 1989, c'est la grande invasion de la micro-informatique tant sur le PC que sur le Mac. Mentionnons les plus connues : Windows de Microsoft, Presentation Manager d'IBM/Microsoft.

En simplifiant quelque peu, on peut dire qu'il y a trois familles d'interfaces graphiques, celles du Macintosh, du groupe d'IBM et de ses compatibles, et des autres systèmes sous Unix. Celle du Mac, la plus ancienne sert de point de référence et se caractérise par l'utilisation obligée de la souris et des icônes. Il suffit de pointer et de cliquer. Celles d'IBM et de ses clones se distinguent par l'utilisation associée de la souris et du clavier au moyen de la barre de menu. Quant aux stations de travail fonctionnant sous Unix, elles seront aussi dotées d'une interface graphique.

Pour bénéficier des avantages de l'intégration, les programmes d'application doivent toutefois être conditionnés à un environnement graphique. Pour cela, il faut convaincre les développeurs de logiciels de proposer des produits compatibles avec ces environnements.

Pour les producteurs d'équipement et d'interfaces, les enjeux sont grands et la compétition est serrée. Macintosh s'efforce de garder son avance de pionnier. Le clan d'IBM/Microsoft et des compatibles propose le système d'exploitation OS/2 et l'interface Presentation Manager. Sous le système d'exploitation Unix, des alliances se nouent : IBM, Digital, HP, et Apollo se liguent et forment l'Open Software Fondation (OSF) afin de définir l'interface graphique Motif. De leur côté, Sun et AT&T proposent l'Open Look. Des indépendants comme NeXT et Metaphor font cavaliers seuls, mais sont aussi courtisés par les grands.

La prédominance d'une interface sur une autre sera moins l'affaire de l'utilisateur final que des grands producteurs et le résultat de leur stratégie de mise en marché. Presentation Manager sera suffisamment près de Windows pour ménager la transition du DOS à OS/2 et les interfaces sous Unix assez ressemblantes pour attirer les adeptes d'OS/2. Macintosh compte sur l'avance de ses milliers de logiciels déjà disponibles, IBM et les compatibles misent sur le parc d'appareils installés dans la grande entreprise et les

3. Pascal Rosier, « Philippe Kahn : le logiciel dépend du matériel », *Micro-systèmes*, no 96 (avril 1989), 115-116.

4. Frank Hayes, « A Guide to GUIs », *Byte*, vol. 14, no. 7 (July 1989), 250-257.

fabricants de stations de travail comptent sur leur position établie dans des créneaux spécialisés, notamment de l'édition électronique et du dessin assisté par ordinateur.

Ces interfaces sont constituées de trois éléments: un module de construction et de gestion de fenêtres, menus et boîtes de dialogue (windowing system) qui constitueront l'interface utilisateur, une interface de programmation des applications (application program interface) qui fait le lien entre les fenêtres et le programme d'application et un générateur graphique (imaging model) qui définit les polices de caractères et les symboles graphiques. Elles peuvent avoir en commun le même système de construction de fenêtres ou utiliser le même générateur graphique. Mais, chacune d'elles dispose en propre d'un « toolkit » ou boîte à outils pour réaliser le lien entre l'application et les éléments graphiques de présentation.

Or, bâtir une interface à l'aide d'une de ces boîtes à outils n'est pas une simple affaire que ce soit avec le Macintosh, sous Windows ou sous Presentation Manager d'OS/2.

En effet, la simplicité d'utilisation finale est bien souvent proportionnelle aux efforts qui ont été fournis pour une conception saine de l'application. A fortiori pour une application graphique, la programmation peut être extrêmement lourde⁵.

C'est ce que confirme le témoignage de deux développeurs d'une société française Priam qui travaillent avec le kit de Windows: « La première étape consiste à consacrer trois mois à la lecture soutenue de trois classeurs de documentation, de respectivement 600, 559 et 304 pages »⁶. En plus de connaître le langage C ou Pascal, il faut appliquer une méthodologie rigoureuse et découvrir les subtilités de la compilation. Sous OS/2, à cet acquis, il faudra ajouter la maîtrise du multitâche et de l'intercommunication entre les tâches. Pour donner une idée de l'importance relative de ces environnements graphiques, Windows possède 400 fonctions, OS/2 en a 700 alors que DOS n'en a que 30. L'auteur de l'article conclut:

À 25 000 F HT le kit de développement OS/2, sans compter les mois d'apprentissage, on comprend que les développeurs rêvent aux pistes de ski, tout en laissant à d'autres le risque de chutes marketing. À terme, OS/2 deviendra cependant... incontournable⁷.

Pour alléger la tâche des programmeurs sous OS/2 et son interface Presentation Manager, de nouveaux outils de développement en interactif font leur apparition sur le marché. Mais, il ne semble pas que ces aides soient d'un secours aussi grand qu'escompté en raison de leur propre complexité et de leur trop grande rigidité, ignorant ou détruisant les codes sources déjà produits à l'aide du kit de Presentation Manager. Sous Windows de tels outils de développement existent déjà et allègent effectivement le travail du programmeur, mais c'est au prix d'une moins bonne performance de l'application.

Les interfaces personnalisées

Si l'interface graphique ne peut être entreprise que par des équipes de développement aguerries, l'interface utilisateur personnalisée sera bientôt accessible à l'amateur grâce à des modules de programmation objet qui lui permettront de dessiner sa propre présentation graphique.

Pour le moment, la principale source des difficultés provient de ce que les langages de programmation traditionnels ne conviennent pas au traitement dans un environnement graphique. Y seront beaucoup plus adaptés les langages orientés objet qui, comme Smalltalk, sont nés avec les premières interfaces graphiques. Les langages traditionnels séparent dans leur écriture les données des fonctions. Les langages orientés objet rassemblent ces données et fonctions pour former des objets qui communiquent entre eux à l'aide de messages. Ces objets peuvent alors être facilement ajoutés, échangés ou modifiés. Appliqué à la réalité de l'interface graphique, les fenêtres, les boutons, etc. deviennent des objets que l'on manipule facilement selon les besoins. Ces langages ont mis du temps à s'imposer, mais maintenant, c'est au tour des langages traditionnels à s'orienter objet. Quick Pascal et Turbo Pascal viennent de paraître respectivement dans des versions orientées objet.

Pour que la tâche des programmeurs soit facilitée, il faudra que les interfaces soient réécrites dans l'un des ces nouveaux langages. C'est ce que se propose de faire Microsoft en transposant Windows en C++, une extension objet du langage C actuel.

Dans la personnalisation de l'interface, Apple innove encore une fois avec Hypercard et son langage Hypertalk, ceux-ci étant conçus à l'intention de l'utilisateur final. Ils tracent la route

5. Stephan Popovitch, « Windows, le standard universel de demain? », *Micro-systèmes*, no 96 (avril 1989), 193-195.

6. Jacques de Schryver, « Programmer sous Windows et OS/2 », *Micro-systèmes*, no 94 (février 1989), 121-129.

7. *Ibid.*, 128.

aux modules en surcouches à l'interface Windows ou PM comme le sera NewWave et donne un avant-goût de ce que sera le cube NeXT de Steve Jobs et son environnement orienté objet Next-Step.

Hypercard, ce logiciel fourni gracieusement par Apple avec le Mac SE, a contribué à faire connaître l'hypertexte dans le grand public. L'hypertexte n'en est pas moins un genre difficile à cataloguer. Il tient à la fois de la base de données, du gestionnaire d'idées, et de l'interface graphique. Mais, avec Hypercard, l'opérateur manipule différents objets : des piles de cartes (boutons, champs pour texte, fonds communs à plusieurs cartes). La conception d'un programme ou pile se fait de façon interactive, à l'aide d'un langage de programmation, Hypertalk. Ce langage, bien qu'il ne soit pas reconnu comme orienté objet à part entière par les spécialistes, retient des notions empruntées à cette nouvelle forme de programmation comme, par exemple, le concept d'héritage par lequel un objet peut « hériter » des propriétés d'un autre.

C'est à l'usage que l'on découvre les applications d'hypertexte. On a qualifié cette nouvelle technique d'« immature technology »⁸ car elle n'a pas encore parfaitement maîtrisé ses méthodes de navigation et de reconnaissance dans un texte. Ce qui ne l'empêche pas de se tailler une place enviable comme interface graphique à une base de textes ou comme intermédiaire donnant accès à un disque optique ou à un CD-ROM. C'est pourquoi l'hypertexte est aussi reconnu comme « un fédérateur d'applications »⁹ et comme un outil de personnalisation mis à la portée de tous.

Sur le PC, NewWave se présente aussi comme un module d'automatisation des procédures et un système de gestion intégré manipulant des objets et établissant des liens entre les applications. Il peut fonctionner par-dessus des interfaces graphiques telles que Windows et Presentation Manager. On en sait peu de choses encore car s'il a été présenté à la presse en novembre 1987 par Hewlett-Packard, il ne fera son apparition sur le marché qu'en juin 1989. Avec NewWave et un concurrent possible, Metaphor apparaît un nouveau concept, celui de surensemble ou surcouches à l'interface graphique standard.

Pour pouvoir véritablement bénéficier des avantages des langages orientés objet, il faudra passer à une nouvelle génération de systèmes. Le NeXT de Steve Jobs est, selon les standards actuels, de la classe des stations de travail avec

ses 8 Mo, son disque optique effaçable de 250 Mo et son système d'exploitation de type Unix. Il bénéficie des avantages de la programmation orientée objet grâce à l'environnement d'exploitation NextStep et au langage de programmation Objective-C. Sa boîte à outils fournit un ensemble d'objets prédéfinis, une trentaine, pour bâtir en interactif des interfaces utilisateurs à l'aide de l'« Interface Builder ». Pour la construction d'un programme, ce module ne dispense pas complètement de la codification, mais allège considérablement le travail du programmeur. Comme le propose Bruce F. Webster¹⁰, les tâches peuvent se partager entre l'utilisateur et le programmeur, le premier dessinant l'interface et le programmeur complétant les liens fonctionnels avec les fichiers créés avec l'Interface Builder.

Les interfaces en langage naturel

Pour simplifier le mode d'emploi du micro-ordinateur, on a fait aussi appel à l'intelligence artificielle, par le recours au langage naturel et à des techniques empruntées aux systèmes experts.

Vers les années 1985, avec l'augmentation progressive de rapidité et de puissance des micros, les grands producteurs de logiciels lancèrent sur le marché les premières interfaces en langage naturel. Apparurent successivement Clout, comme programme associé au SGBD RBase, The Intelligent Assistant, comme module d'interrogation du gestionnaire de fichier Q&A et Hal, comme utilitaire en mémoire résidente de Lotus 1-2-3. Ces produits d'interface de bases de données, selon Eric Seiden, « procèdent le plus souvent par analyse syntaxique suivie d'une représentation sémantique de type prédicat »¹¹. Douées d'un vocabulaire de 300 à 500 mots, et de règles de production empruntées aux langages de l'intelligence artificielle, ces interfaces d'interrogation permettent de formuler les questions dans un langage quasi naturel mais limité au domaine de connaissance de la base.

Ainsi, une base de données sous Q&A peut être interrogée en tapant au clavier la question dans les mots qui viennent spontanément à l'esprit. Au préalable, il aura fallu structurer la base de la façon classique : nom et longueur des champs, nature des données, etc. ; il aura fallu également instruire l'« Assistant intelligent », au moyen d'une série de huit courtes leçons, des termes utiles à connaissance du contenu de la base. À l'interrogation, l'Assistant intelligent analysera la

8. Janet Fiderio, « A Grand Vision », *Byte*, vol. 13, no. 10 (October 1988), 237-244.

9. « Hypertexte : l'outil de navigation documentaire », *Micro-systèmes*, no 94 (février 1989), 105-113.

10. Bruce F. Webster, *The NeXT book*, New York, Addison-Wesley, 1989, p. 338.

11. Eric Seiden, « L'ordinateur comprend ce qu'il peut », *L'Ordinateur individuel*, no 85 (octobre 1986), 102-108.

question puis proposera sa propre formulation par un énoncé plus structuré. Si l'opérateur est d'accord avec le sens de la question, il tapera une touche au clavier et la recherche débutera ; sinon, la question devra être formulée autrement. Si en cours d'analyse de la question, un terme lui est inconnu ou ambigu, l'Assistant demandera des précisions qu'il gardera en mémoire après en avoir demandé l'autorisation.

Les concepteurs du système avaient établi, entre autres critères de design, la possibilité d'une formulation de la question dans une forme spontanée, la capacité de contre-vérification par l'opérateur du sens de la question et la possibilité d'une mise à jour synchronisée du vocabulaire avec la base correspondante. Quant à l'interface permettant l'apprentissage de l'Assistant intelligent, elle se devait d'être la plus naturelle et la moins fastidieuse possible pour éviter de frapper ce que les spécialistes appellent « le mur du traitement du langage naturel »¹². Il ne faut pas que l'étape d'apprentissage préalable nécessite l'utilisation d'une interface utilisateur plus artificielle que celle que l'on tente d'éviter.

Ces interfaces en langage naturel sont moins universelles que les interfaces graphiques. Elles sont comme les correcteurs d'orthographe des traitements de textes liées à la langue qu'elles « parlent ». Q&A et son interface ont été traduits et adaptés dans six langues : en allemand, en suédois, en français, en néerlandais, en italien et en finnois. Q&A devient en français, par la force des choses, « Q&R » pour Question et Réponse.

Ces interfaces en langage naturel, en dépit de leur ingéniosité n'ont pas encore gagné la faveur du grand public. Attrayantes pour le nouveau venu, elles sont trop encombrantes pour l'habitué d'un langage par commandes. Elles sont aussi encombrantes en termes d'espace et de puissance que le graphisme et la souris. Q&R et son Assistant intelligent exigent 512k de mémoire.

Présentées avec le même éclat qu'aujourd'hui la programmation objet, ces interfaces en langage naturel ont cédé le pas dans l'actualité informatique aux langages structurés d'interrogation SQL (Structured Query Language). Ces langages de requête SQL, qualifiés d'espéranto par certains, jouent de plus en plus le rôle de langue commune d'accès aux bases de données relationnelles et de moyen de communication entre petits et gros ordinateurs. Par un curieux retour des choses,

on retrouve le concept de formulation en langage naturel et de traduction automatique dans une forme structurée semblable à celle de l'Assistant Intelligent, mais cette fois sur le VAX avec le programme EasyTalk qui analyse la question exprimée en langage naturel et la transcrit en SQL, le langage de requête utilisé par la base de données sous Oracle.

Les interfaces que l'on ne voit pas

Comme l'affirme Eric Sebbag de *Micro-systèmes*, « la meilleure interface est sûrement celle que l'on ne voit pas »¹³. Interrogé par cette même revue, Philippe Kahn, le directeur de Borland confirme : « La convivialité vient de l'« intérieur » du logiciel bien plus que de son interface... Ce qu'il faut c'est que le logiciel comprenne les intentions de l'utilisateur »¹⁴.

En voulant augmenter l'éventail des fonctions disponibles et la puissance des logiciels, les producteurs en ont augmenté la complexité. Les traitements de texte sont devenus des logiciels d'édition électronique, les questionnaires de fichiers ont imité les SGBD relationnels.

Les éditeurs de SGBD relationnels, à l'inverse, ont éprouvé le besoin de rendre leurs produits plus accessibles en masquant la complexité du relationnel et en simplifiant l'interface utilisateur. Pour concilier puissance et facilité d'utilisation, ils ont multiplié les menus déroulants, ajouté des fenêtres « pop-up », allégé les procédures de liaisons de fichiers et rendu des opérations transparentes telles que la sauvegarde de la base avant la restructuration des fichiers.

Ils ont aussi amélioré le langage de communication avec la base de données tant au moment de l'implantation que de sa consultation. Ainsi, dBase est passé du simple point sur l'écran au mode Assistance avec menus affichés, du langage d'interrogation par commande au langage structuré SQL. De son côté, Paradox a proposé le langage de requête par l'exemple (QBE) où les fichiers sont représentés schématiquement et les liens entre eux par un terme commun affiché en surbrillance.

Jim Seymour du *PC Magazine* a recueilli les suggestions d'utilisateurs et en a fait part dans une de ses chroniques¹⁵. Selon Seymour, une interface doit également correspondre aux habitudes de travail de l'utilisateur. Il doit être

12. Gary G. Hendrix and A. Walter Brett, « The Intelligent Assistant », *Byte*, vol. 12, no. 14 (December 1987), 251-258.

13. Alain Baritault et Eric Sebbag, « Une révolution au quotidien », *L'Ordinateur individuel*, no 115 (juin 1989), 92-102.

14. Pascal Rosier, « Philippe Kahn... », 116.

15. Jim Seymour, « Better Software Interfaces », *PC Magazine*, vol. 6, no. 20 (November 24, 1987), 77-78.

possible de personnaliser facilement les menus pour ajouter l'appel d'un programme ou éliminer tel autre qu'on n'utilise jamais et qui encombre le menu. Ce voeu a été réalisé avec le traitement de textes Sprint qui propose l'environnement de son choix, à la Wordstar ou à la WordPerfect ou d'en créer un de son cru.

On peut exiger d'une interface que non seulement elle mette en relation l'opérateur et le programme, mais qu'elle réagisse aux commandes comme il le propose dans sa chronique. Que les logiciels soient dotés d'un mode d'apprentissage pour abréger les opérations répétitives à la façon d'Excel. On pourrait en proposer d'autres aussi. Que des écrans d'aide « dans le contexte » soient disponibles au moment de la difficulté rencontrée. Qu'après trois tentatives infructueuses, un menu de dépannage apparaisse automatiquement comme c'est le cas avec VP-Expert. Que des modules d'aide intelligents ou mini systèmes experts soient incorporés aux programmes comme il en existe un avec le logiciel d'édition électronique PageMaker...

La plupart des générateurs de systèmes experts peuvent maintenant faire appel à des programmes extérieurs pour manipuler des données externes et servir d'interface intelligente à des fichiers de format 1-2-3 ou dBase. Mais l'intégration entre une base de données et une base de connaissances peut aller plus loin. Le plus intéressant reste à venir car, dans le parler de la micro, poussent des hybrides du croisement des systèmes experts, des bases de données et de l'hypertexte. La revue *DBMS*¹⁶ décrivait dans un récent numéro une expérience de croisement d'Oracle, un puissant SGBD relationnel et de l'HyperCard de Macintosh pour obtenir la capacité de traitement de l'un et la convivialité de l'autre.

Grâce à un logiciel comme Nexpert-Object, qui intègre dans une même structure base de données, système expert et hypertexte, il est possible, par exemple, d'incorporer les procédures et politiques aux données d'exploitation d'une entreprise et d'extraire automatiquement des indicateurs et résultats d'analyse à la lumière de ces politiques. Au dire de Dave Thomas¹⁷, chercheur en programmation orientée objet de Carleton University, les SGBD orientés objet, dont Gemstone sur VAX et sur la station de travail Sun est le prototype, seront des produits courants sur le marché au cours des années 1997.

Mais nous sommes là aux confins de la micro-informatique. Pour « aller au-delà du relationnel », selon l'expression de Claire Rémi¹⁸, il faut pénétrer dans les laboratoires de recherche en informatique et avoir accès aux gros ordinateurs. Après une première génération de SGBD inspirée des modèles hiérarchique et réseau et la génération actuelle, issue du modèle relationnel, une troisième est en émergence. Il s'agit des SGBDC ou gestionnaires de bases de connaissances associant le modèle objet servant à décrire non seulement des graphiques mais aussi des connaissances. Avec de tels outils, on pourra concevoir des interfaces intelligentes qui iront, sinon au devant de l'intention de l'utilisateur, du moins suivront son cheminement. Il est permis de rêver à ce « Conseiller intelligent » qui aidera le bibliothécaire de référence à répondre de façon précise à des questions floues...

L'interface utilisateur et le monde de la documentation

Le monde de la documentation pourrait bien se contenter, au cours des prochaines années, de l'interface textuelle et réserver la puissance de ses micros au traitement des données. La description bibliographique, l'analyse documentaire et même le plein texte peuvent se passer de l'affichage graphique, et à un degré moindre, du langage naturel, car les bases documentaires sont essentiellement textuelles. À la rigueur, les systèmes pourraient même rester fidèles au système d'exploitation actuel et aux équipements de bas de gamme soit les PC-Xt et AT. D'autant plus que la barrière du 32 Mo est levée avec la version 4.0 du DOS. Ainsi, le monde de la documentation resterait à l'écart de cette course à la puissance et au graphisme, qui n'est pas sans mirage. Il minimiserait ainsi les coûts d'équipement et de développement et augmenterait les chances pour les petits centres de documentation et les petites bibliothèques d'accéder à l'informatique documentaire. C'est là une stratégie possible de développement.

Mais, l'informatique documentaire va probablement suivre, par un effet d'entraînement, le mouvement de la micro-informatique générale et passer au mode graphique. L'invitation est d'autant plus grande qu'il y a de plus en plus de produits documentaires disponibles sur le Mac. Elle sera aussi attirée par les possibilités de multi-tâche et de l'intertâche sous OS/2 et en plus du

16. Steve Roti, « ORACLE meets Hypercard », *DBMS*, vol. 2, no. 1 (January 1989), 52-62.

17. David Thomas, « Object-oriented Databases and Persistent Objects », *Journal of Object-oriented Programming*, vol. 2, no. 2 (July/August 1989), 50-60.

18. Claire Rémi, « Les SGBD : au-delà du relationnel », *Micro-systèmes*, no 97 (mai 1989), 175-180.

multi-poste sous Unix. Si l'informatique documentaire franchit à son tour le mur de l'interface graphique, elle devra s'attendre à en payer le prix par l'acquisition de micro-ordinateurs plus puissants et de logiciels plus coûteux. Peut-être, devrait-elle dès maintenant recycler son équipement dans le traitement de textes et s'équiper de micro-ordinateurs rapides de haut de gamme...

Les bibliothécaires ont pris conscience de l'importance de l'interface utilisateur moins par la micro-informatique que par les catalogues en ligne accessibles au grand public. Dans leur passage de la fiche 3x5 au dossier électronique, ils ont d'abord imité les grands serveurs de bases de données en élargissant l'éventail des clés d'accès et en proposant des fonctions de recherche similaires. Les catalogues de bibliothèques s'ouvrent maintenant à d'autres sources d'information: les tables des matières, les dictionnaires et les bases de données sur CD-

ROM. Les concepteurs de systèmes de gestion de bibliothèques ont élaboré des lignes directrices pour la présentation d'écrans et la disposition des menus et de l'information affichée¹⁹. Sur les gros systèmes, l'affichage demeurera en mode texte pour un temps encore, l'interface graphique sur un ordinateur central auquel est branché un grand nombre de terminaux est trop lourde à gérer. Mais, on peut penser que le micro-ordinateur, en prenant le relais du système central, sera en mesure de jouer le rôle de station frontale graphique. C'est à lui que reviendra la tâche d'ajouter aux trop sévères catalogues de bibliothèque l'attrait et l'intelligence.

Gérard Mercure

École de bibliothéconomie et des sciences de l'information
Université de Montréal

19. Joseph R. Matthews, «Suggested Guidelines for Screen Layouts and Design of Online Catalogs», *Library Trends*, vol. 35, no. 4 (Spring 1987), 555-570.

NOTE DE LA RÉDACTION

Dans la publication du volume 35, numéro 2, avril-juin 1989 de notre revue, une erreur s'est malencontreusement glissée dans la conclusion de l'article de Pierre Pelou intitulé «Télécommunications et mémoires optiques dans la documentation internationale».

Nous nous en excusons auprès de l'auteur et de nos lecteurs et nous reproduisons ici le texte tel qu'il aurait dû se lire.

Conclusion

La documentation internationale est aujourd'hui le théâtre d'une réflexion nouvelle qui, à partir de sites éloignés, tente de présenter des formes variées de traitement et de consultation de l'information, soit en local, soit à distance. Télécommunications et mémoires optiques apportent une alliance nouvelle à l'approche documentaire. Les professionnels de l'information doivent être les acteurs privilégiés de ce changement et sont particulièrement bien placés pour apporter aux techniciens comme aux politiques les stratégies nécessaires de produits définis plus étroitement en fonction de l'usage réel et ciblé.

Index des affaires

UN RÉPERTOIRE QUI VOUS FERA GAGNER DU TEMPS!

Le seul répertoire bibliographique portant exclusivement sur le monde des affaires et la vie économique du Québec.

Un instrument de recherche documentaire qui permet un choix judicieux d'articles parmi les principales publications d'affaires québécoises (*Les Affaires*, *Finance*, *PME*, *Commerce*, *This week in business*, *Affaires +*, etc.)

L'INDEX DES AFFAIRES permet entre autres de:

- constituer des dossiers sur des entreprises;
- suivre l'évolution d'un secteur économique;
- analyser un marché;
- identifier des clients ou fournisseurs potentiels;
- mieux gérer ses finances personnelles; etc.

De consultation simple et rapide, l'INDEX DES AFFAIRES offre 12 000 articles indexés par année publiés sous forme de 10 mensuels et d'une refonte annuelle.

Pour abonnement ou renseignements:

Inform II
Microfor

4899 Ste-Catherine ouest, suite 430, Westmount, QC
H3Z 1T3 (514) 464-5951

Les spécialistes en édition de base de données