

L'informatique juridique : en progression vers un processus d'intelligence artificielle

Jean Goulet

Volume 21, numéro 3-4, 1980

La rédaction des lois

URI : <https://id.erudit.org/iderudit/042407ar>

DOI : <https://doi.org/10.7202/042407ar>

[Aller au sommaire du numéro](#)

Éditeur(s)

Faculté de droit de l'Université Laval

ISSN

0007-974X (imprimé)

1918-8218 (numérique)

[Découvrir la revue](#)

Citer cet article

Goulet, J. (1980). L'informatique juridique : en progression vers un processus d'intelligence artificielle. *Les Cahiers de droit*, 21(3-4), 615-671.

<https://doi.org/10.7202/042407ar>

Résumé de l'article

This paper deals primarily with computer-assisted legal research. It attempts to sketch the current state of the art, mainly in the United States and Canada, with special reference to systems oriented towards the processing of legislative data. The author suggests a checklist of the main requirements the systems of the 80's will have to answer to, in order to fulfill the growing needs of the new computer-minded generations of law graduates.

Along these lines, this paper deals also with the second generation systems dedicated to automated legal research ; these could be expected to show some form, albeit elementary, of humanlike intelligence. Four prototypes of such systems are considered; they are the American Bar Foundation's and Jeffrey Meldman's systems, as well as the well-known JUDITH and TAXMAN systems.

The paper concludes on a glimpse of the Third Wave of computerized legal research, in the belief that the legal profession will meet the challenge of the computer age, will learn to live and work with this new technology, and will master the artificial but sometimes acute intelligence of our new friend, the Robot.

L'informatique juridique : en progression vers un processus d'intelligence artificielle

Jean GOULET *

This paper deals primarily with computer-assisted legal research. It attempts to sketch the current state of the art, mainly in the United States and Canada, with special reference to systems oriented towards the processing of legislative data. The author suggests a checklist of the main requirements the systems of the 80's will have to answer to, in order to fulfill the growing needs of the new computer-minded generations of law graduates.

Along these lines, this paper deals also with the second generation systems dedicated to automated legal research; these could be expected to show some form, albeit elementary, of humanlike intelligence. Four prototypes of such systems are considered; they are the American Bar Foundation's and Jeffrey Meldman's systems, as well as the well-known JUDITH and TAXMAN systems.

The paper concludes on a glimpse of the Third Wave of computerized legal research, in the belief that the legal profession will meet the challenge of the computer age, will learn to live and work with this new technology, and will master the artificial but sometimes acute intelligence of our new friend, the Robot.

	<i>Pages</i>
Introduction	616
1. L'intelligence d'informatiser : l'apprentissage d'un processus	618
1.1. L'apprentissage de l'informatique	619
1.1.1. Les premiers acquis par le texte intégral	620
1.1.2. Les premières diversifications de l'acquis	622
1.1.3. Acquérir la diversification	623

* Professeur titulaire, Faculté de Droit, Université Laval.

	<i>Pages</i>
1.2. La maîtrise de l'informatique	624
1.2.1. La structuration en réseau	626
1.2.2. L'hétérarchisation des systèmes	627
1.2.3. La jonction des chaînes documentaires	627
1.2.4. L'indexation différée	628
1.2.5. Des logiciels traducteurs	628
1.2.6. Des systèmes interactifs	629
1.2.7. Des systèmes accessibles	630
1.2.8. Des systèmes de saine ergonomie documentaire	632
1.2.9. Des systèmes sympathiques	634
1.2.10. Des systèmes économiques	635
2. Informatiser l'intelligence : le processus d'apprentissage	640
2.1. L'apprentissage par la logique	641
2.1.1. La normalisation des messages	642
2.1.2. La structuration des processus	646
2.1.2.1. Le processeur ABF	646
2.1.2.2. JUDITH	650
2.1.2.3. MAC	654
2.1.2.4. TAXMAN	659
2.2. La logique de l'apprentissage	665
2.2.1. Les conditions du système	666
2.2.2. Le projet de système	668
Conclusion	670

Introduction

Il y a plus de vingt ans aujourd'hui que l'informatique juridique a pris sa place dans les mondes conjoints du droit et de la documentation automatisée¹.

On se souvient que la formation de cette nouvelle discipline avait été suggérée par Lee Loevinger en 1949², soit quelques années seulement après

1. On ne saurait déterminer avec une exactitude rigoureuse qui fut le premier juriste à travailler avec l'ordinateur. Il se pourrait que ce fut toutefois, le professeur Morgan de l'Université d'Oklahoma, actif dans ce domaine dès 1957; voir nos remarques à ce sujet dans « Revalorisation du droit et jurimétrie », (1967-68) 9 *C. de D.* 9-36 à la page 15. Par ailleurs, au Canada, monsieur Guy Forget, de l'Université Laval à Québec, travaillait dès l'été 1964 à l'automatisation de la recherche en droit des assurances à partir d'un sélecteur optique Miracode. Ce prototype de système suscita l'intérêt des chercheurs qui lancèrent MØDUL à l'été 1967, un contemporain de DATUM (de l'Université de Montréal) dont les travaux débutèrent au même moment.

En France, il faut souligner aussi la clairvoyance d'un Lucien Mehl qui parlait d'informatique il y a déjà vingt ans; *vide*: « Les sciences juridiques devant l'automatisation »; (1960) 3 *Cybernetica* 22.

2. L. LOEVINGER, « Jurimetrics: "The Next Step Forward" », (1949) 33 *Minnesota Law Review*, 455-493.

que la nouvelle science cybernétique eut été lancée elle-même à Paris par l'Américain Norbert Wiener³. Coïncidence ou simple signe des temps? On ne le sait pas vraiment! Un fait cependant reste indéniable, c'est également à cette époque que les textes législatifs, principaux ou subordonnés, ont commencé à se multiplier à un rythme tel que George Ripert lui-même en concluait déjà alors au « déclin du droit »⁴. Or, la prolifération des textes légaux de toutes origines n'a pas diminué ensuite! Il y a au moins deux millions d'articles de droit statutaire en vigueur aux États-Unis à l'heure actuelle et sûrement plus de trois millions de causes rapportées⁵. En Angleterre, le gouvernement libéral réformateur de 1911 avait publié 450 pages de législation, alors que le législateur de 1975 a édicté quelque 3 000 pages de législation directe et 10 000 de législation subordonnée. Au Québec, on estime que nos tribunaux rendent quelque 250 000 jugements chaque année dont quelque 1 024 ont été retenus pour publication sous forme au moins résumée dans diverses publications de SOQUIJ⁶.

Si l'on en croit les théoriciens de l'information George H. Miller ou Abraham Moles par exemple, il vient un moment où le surcroît de renseignements émis ne peut plus être accepté par son récepteur désigné⁷.

3. *Cybernetics or control and communication in the animal and the machine* a été terminé en novembre 1947 et publié peu après chez Hermann à Paris. L'ouvrage, vite remarqué, fait ensuite l'objet d'une recension critique signée par un professeur de l'Institut Catholique de Paris, le père Dubarle, dans le numéro du 28 février 1948 du journal *Le Monde*. Rappelons enfin que la revue *Esprit* consacrait quelque 59 pages de son numéro de septembre 1950 aux *machines à penser* (voir : L. DELPECH, « Un créateur : Norbert Wiener » in *Le dossier de la cybernétique*, Bruxelles, Éd. Marabout, 1968, pp. 29-45); *id.*, *La cybernétique et ses théoriciens*, Bruxelles, Casterman, 1972.

Cybernetics a été publié à New York en 1948.

4. G. RIPERT, *Le déclin du droit*, Paris, L.G.D.J., 1949.

5. Nous extrapolons tout simplement ici des chiffres déjà rapportés dans « Revalorisation du droit... », *op. cit.*, *supra*, note 1.

6. Ces chiffres sont rapportés par Me Philippe Miller, l'ancien arrêtiiste du Barreau du Québec aujourd'hui collaborateur à SOQUIJ dans : « Information juridique : l'arrêtiisme, dix ans plus tard », (1980) 40 *R. du B.* 164.

7. Nous faisons face ici à des problèmes reliés de perception et de décodage de l'information. Il semble aussi que le cerveau ne puisse absorber guère plus de 16 bits/sec. (selon A. MOLES, *Art et ordinateur*, Bruxelles, Casterman, 1971). La limite de perception du récepteur apparaît donc bien évidente.

Même si, en théorie de l'information pure, la signification est dissociée de l'information, il devient évident que le décodage permettant d'atteindre cette signification suit le processus de perception original, l'action psychique suivant nécessairement le processus physiologique (voir aussi : A. MOLES, *Théorie de l'information et perception esthétique*. Paris, Denoël, 1972; G. H. MILLER, « The Magical Number Seven, Plus or Minus Two: Limits on Our Capacity for Processing Information », in *The Psychology of Communication*, Middlesex Eng., Basic Books, 1967).

Le seuil de l'intelligibilité est franchi ! Tout le système se désorganise lui-même par sa propre production. L'information⁸ devient entropie, cette quantité de dégradation du système⁹ qu'avait intuitivement perçue Ripert à propos du Droit et que subissent hélas maintenant, mais à leur corps défendant, les juristes du monde actuel.

C'est donc pour contrôler, réorganiser et restructurer les problèmes de macro-recherche documentaire que l'ordinateur fut appelé à la rescousse à la fin des années '50 par des chercheurs de disciplines diverses. Les juristes n'ont pas échappé alors au mouvement et ont contribué au contraire à implanter des systèmes efficaces¹⁰, sur lesquels nous vous proposons aujourd'hui de faire d'abord le point, en les observant en premier lieu, mais en les critiquant également ensuite. À partir de ces réflexions, nous poserons après une question plus importante peut-être, soit celle qui porte sur les développements futurs des systèmes documentaires que nous connaissons déjà. Nous allons élaborer dès lors, sur le thème des logiques formelles en un premier temps, mais nos propos s'arrêteront surtout en réalité sur les systèmes informatiques dits intelligents, en observant quelques exemples de ces nouvelles machines à penser qui ont des juristes pour auteurs.

Engageons-nous donc maintenant sur la voie des recherches qui nous permettront de changer de signe l'entropie informative actuelle, en suivant l'évolution des systèmes traitant l'informatique juridique depuis ASPEN jusqu'à TAXMAN.

1. L'intelligence d'informatiser : l'apprentissage d'un processus

Il n'est pas de notre propos de présenter ici à nous-même un historique de l'informatique juridique qui a été maintes fois repris un peu partout dans l'abondante littérature qui traite de ce sujet¹¹. Aussi nous contenterons-nous

8. Suivant la ligne de pensée amorcée à la note précédente, nous entendons ici par *information* un phénomène identique à celui que décrit Louis Couffignal lorsque ce dernier associe *action physique* et *action psychique*, *support* et *sémantique* (L. COUFFIGNAL, *La Cybernétique*, Paris, P.U.F., 1968 (Coll. Que sais-je?, n° 638), p. 32; aussi R. RUYER, *La cybernétique et l'origine de l'information*, Paris, Flammarion, 1956, p. 11).

9. Caricaturant les définitions, on pourrait dire que l'entropie est la « mauvaise » information, celle qui désorganise ce que la « bonne » information avait structuré (*vide*: J. GUILLEMAUD, *Cybernétique et matérialisme dialectique*, Paris, Éditions Sociales, 1965, pp. 138 et sq.; A. MOLES et al., in *La Communication*, Paris, Denoël, 1971). Albert Ducrocq définit d'ailleurs l'information comme la *neg-entropie* dans *Découverte de la cybernétique*, Paris, Julliard, 1955, p. 221 (au même effet: GUILLEMAUD, *op. cit.*, p. 155).

10. Comme nous le verrons plus loin, tout commence évidemment par John Horty!

11. Nous nous contenterons de renvoyer ici le lecteur à notre article précité « Revalorisation du droit... », à note I, et à nos commentaires dans « La machine et le droit et la machine du

de simplement rappeler quelques systèmes anciens et quelques nouveaux venus, non pas d'ailleurs dans un but simple de retour en arrière, mais plutôt pour tenter de suivre l'évolution de cette discipline, principalement depuis les dix dernières années.

Notre hypothèse voudrait ici que si nous n'arrivions pas à déceler de progrès significatifs, il nous faudrait conclure que cette forme de traitement a atteint un plateau de perfection tel qu'on ne pourrait plus en attendre de transformations véritables, ou, encore que le développement de cette technique documentaire se heurte à un obstacle si formidable, qu'il faudrait d'abord le vaincre pour continuer notre progression. Cette démarche devrait cependant faire alors l'objet d'une étude spéciale pour connaître les coûts et la rentabilité de cette nouvelle forme de progrès. Il nous faudrait en fait déterminer alors la somme d'efforts que ces améliorations exigeraient, et nous demander si un tel investissement en vaudrait la peine.

Amorçons donc tout de suite notre étude, en commençant, il va sans dire, par le système déterminant, celui de qui presque tous les autres sont tributaires de quelque façon, soit le système ASPEN de John Horty¹².

1.1. L'apprentissage de l'informatique

Bien connu de tous, ce système ne requiert pas qu'on en explique profondément les rouages internes. ASPEN a eu le mérite inestimable d'être le premier à retenir le texte intégral des bases de données traitées et d'y effectuer un repérage d'informations subséquent en un mode booléen, par la combinaison de mots-clés joints ou disjoints par les opérateurs ET, OU ou SAUF. Ces mots clés, comme on le sait, s'ils appartiennent à la base de données, sont adressés en fonction d'un document quelconque, et ils en appellent le texte intégral, ou la référence, suivant les instructions de la requête qui la contient. Ainsi opérait ASPEN, ainsi fonctionna ensuite LITE (et son successeur FLITE)¹³, ainsi ont été conçus, en termes plus ou moins

-
- droit », (1973) 14 *C. de D.* 412-498 (474, note 2). On pourra consulter également L. SHULTE, « A Survey of Computerized Legislative Information System », (1979) 72 *Law Library Journal*, 99-129; P. NYCUM, « Law and Computers: overview update 1975 », (1975) 68 *Law Library Journal*, 234-253; D. T. LINK, « Law Searching by Computer », in *Automated Law Research*, Chicago, American Bar Association, 1972, pp. 3-9; R. C. LAWLOR « Historical Perspective and Future Prospects » in *Sense & Systems in Automated Law Research*, Ronald A. May (ed.), Chicago, American Bar Association, 1975, pp. 5-34.
12. On a beaucoup publié sur John Horty, mais peu de textes émanent directement de l'auteur d'Aspen. L'un des rares est « Use of the Computer in Statutory Research and the Legislative Process », in *Computer and the Law*, R.P. Bigelow (ed.), New York, C.C.H. 1966, pp. 48-55.
13. On pourra consulter sur ce système, *inter alia*, les numéros spéciaux de *United States JAG Law Review*, publiés en 1966 et 1972.

analogues, presque tous les autres systèmes créés entre 1965 et 1970, en Amérique du Nord et en Europe.

1.1.1. Les premiers acquis par le texte intégral

La recherche documentaire, en texte intégral, a été de la sorte retenue pour la législation par MØDUL¹⁴ au Canada et DOCILIS¹⁵ en France, par les États de Washington et d'Illinois, par GIPSY¹⁶ de l'État d'Oklahoma et maintenant par JURIS¹⁷ du Ministère fédéral américain de la Justice. En jurisprudence, un tel choix sous-tend également la règle d'action générale. DATUM¹⁸ jusqu'à sa disparition, opérait de même. OBAR, devenu LEXIS¹⁹, a choisi également cette voie. Même QUIC/LAW²⁰ est fondamentalement booléen, même si son logiciel permet au surplus une pondé-

-
14. Voir J. GOULET, S. HOULE et J. LECLERC-HOUDE, « Jurimétrie et Loi : MODUL », (1971) 6 *R.J.T.*, 116-132; à propos de ce système, on pourra consulter également : D. LE MAY, « Pour un index du droit en vigueur », (1977) 18 *C. de D.*, 733; *id.*, « L'informatique juridique au Québec : état de la question », (1978) 19 *C. de D.*, 987.
 15. Nous renvoyons ici le lecteur à notre bibliographie en note 12 de « La machine et le droit... », *supra*, note 11.
 16. Voir : A. L. THOMAS. « GIPSY » : General Information Processing System for Oklahoma », in *Automated Law Research*, *supra*, pp. 119-124.
 17. Voir : B.W. BASHEER, « JURIS : Justice Retrieval and Inquiry System », in *Automated Law Research*, *supra*, 55-65; G. S. KONDOS, « JURIS : Remote Terminal Legal Information Retrieval at the United States, Department of Justice », (1971) 4 *Law and Computer Technology*, 147-155.
 18. On pourra consulter sur ce système deux numéros spéciaux de la *Revue Juridique Thémis* publiés en 1971 et 1976. En juin 1979, le Conseil d'Administration de SOQUIJ a décidé de ne pas poursuivre l'exploitation commerciale de DATUM dont la base de données reste néanmoins disponible au chercheur qui désirerait étudier DATUM et, aussi, pour SOQUIJ même qui en tire à l'occasion une exploitation appréciée pour ses fins de recherche interne.
 19. On a beaucoup publié sur LEXIS, mais nous nous contentons de renvoyer ici le lecteur aux derniers textes parus, dont : M. F. DEE and R. M. KESSLER, « The Impact of Computerized Methods on Legal Research Courses : A Survey of LEXIS Experience and Some Probable Effects of WESTLAW », (1976) 69 *Law Library Journal*, 924-931; F. M. GREGURAS, « The Eyes of Texas Are Upon LEXIS : Computer Assisted Legal Research », (1976) 17 *South Texas Law Journal*, 349-365; J. A. SPROWL, « Computer Assisted Legal Research — An Analysis of Full-Text Document Retrieval Systems, Particularly the LEXIS System », (1976) *American Bar Foundation Research Journal*, 174-225; R. LAWRENCE, « Introducing Students to LEXIS : A Model of Self-Teaching Exercise », (1979) 71 *Law Library Journal*, 467-470; J.S. RUBIN, « LEXIS : An Automated Research System », in *Automated Law Research*, *supra*, note 11, p. 35-41. R.M. MCGONIGAL, « Implementation and Cost Effectiveness of Computerized Legal Research — LEXIS and WESTLAW compared » (1978) 1 *Computer/Law Journal*, 359-378.
 20. Voir : H. LAWFORDE, « QUIC/LAW : Project of Queen's University », in *Automated Law Research*, *supra*, note 11, pp. 67-83; aussi « Queen's University QUIC/LAW Project », (1974) 7 *Law and Computer Technology*, 63-70; H. LAWFORDE, « QUIC/LAW Ready to Test Retrieval in Lawyers' Office », (1972) 4 *Canadian Data Systems*, 40.

ration des documents reliés à la fréquence d'occurrence des mots clés à l'intérieur des documents référés.

WESTLAW²¹ lui-même, un descendant direct de QUIC/LAW, ne fait pas entièrement exception à la règle générale, puisqu'on y repère ses documents, des résumés d'arrêts il est vrai!, à partir aussi de techniques quasi booléennes, auxquels on peut joindre une recherche par termes numériques suivant le système de « clés » West, si on le désire.

En vérité, il ne faut pas chercher longtemps pour réaliser que presque tous les grands systèmes de recherche documentaire automatisée présentement opérationnels fonctionnent à partir de textes intégraux (qui peuvent être cependant des résumés, comme c'est le cas pour WESTLAW) dont les termes composants constituent les mots clés dont le chercheur se servira ensuite pour amener en sortie les documents pertinents à son interrogation. D.P.S.²² et STAIRS²³ ont fait école, et Aspen Search IV d'Aspen Systems, et SIRS de Data Retrieval²⁴, LEXIS de Mead Data, Q/L de Q/L Systems ou Sagittaire²⁵ de l'Université Laval sont leurs descendants.

-
21. WESTLAW est opéré par le prestigieux éditeur West Publishing de St-Paul, Minnesota. Ce système fonctionne à partir du logiciel conçu et écrit par QUIC/LAW à l'Université Queen's de Kingston, Ontario par l'équipe du professeur Hugh Lawford. On pourra consulter sur ce système : M. DEE and R. R. KESSLER, *supra*, note 19; A. O. GINNOW, « The West Computer-Assisted Legal Retrieval System », (1975) 8 *Law and Computer Technology*, 82-86; J. A. SPROWL, « The WESTLAW System — A Different Approach to Computer Assisted Legal Research », (1976) 16 *Jurimetrics Journal*, 142-148; K. A. ZICK, « Developing and Implementing a Law School WESTLAW Orientation Program », (1979) 72 *Law Library Journal*, 260-275; R.M. MCGONIGAL, *supra*, note 19.
 22. Le *Document Processing System* a été mis en marché pendant plusieurs années par son auteur, la société IBM, qui en faisait alors son logiciel du repérage d'information conçu en fonction de ses ordinateurs de la série 360. Le CEDIJ à Paris, MØDUL/DEPLOI à Québec et le législateur de l'État de Washington en ont fait à l'époque un usage abondant.
 23. STAIRS (Storage and Information Retrieval System) a succédé à D.P.S. lorsque les ordinateurs IBM de série 370 ont succédé à ceux de série 360. Du jeu global des logiciels qui forment l'ensemble STAIRS, c'est le sous-ensemble AQUARIUS qui est le plus important parce qu'il aboutit ultimement à offrir à son utilisateur toutes les possibilités d'effectuer un travail à distance (M.P.H. DAVIES, « Introduction to STAIRS: a Legislative and Administrative Information National System », in: *Why Legislators Need Data Processing*, Bruxelles, IBM Public Sector Industry Center, 1976; S. E. FURTH, « STAIRS: An Interactive Full-Text Retrieval System », in *Automated Law Research*, *supra*, note 11, pp. 19-33.
 24. A. FROST, « SIRS, ALTER and TYPE: Three Integrated Text Processing Systems », in *Automated Law Research*, *supra*, note 11, pp. 125-129;
 25. *Sagittaire* est un logiciel consacré fondamentalement au repérage d'information et mettant à profit les avantages du langage APL. Il a pour auteur principal François Joncas du Centre de Traitement de l'Information de l'Université Laval à Québec. La banque des *Lois refondues du Québec* (1977) sera bientôt traitée entièrement en *Sagittaire*.

Dès lors, il nous faut conclure que les dix dernières années n'ont pas apporté de profond renouvellement dans toute cette machine, pourtant encore fort complexe, de la recherche documentaire. Les modifications les plus nombreuses se sont situées depuis plusieurs années aux plans de la souplesse des logiciels qui sont devenus interactifs et dont la structure interne des fichiers s'est aussi améliorée avec le temps. En fait, deux changements apparaissent particulièrement notables pour l'utilisateur des systèmes dont on se préoccupe d'ailleurs maintenant en priorité, plutôt que de donner la vedette, comme c'était le cas jadis, aux systèmes eux-mêmes.

C'est ainsi que depuis quelques années, l'accès aux systèmes est devenu facile, alors que les banques de données traitées se sont élargies ou que les chaînes documentaires se sont allongées.

1.1.2. Les premières diversifications de l'acquis

L'informatique juridique n'a pas d'avenir tant et aussi longtemps qu'elle laisse croire à ses bénéficiaires éventuels qu'ils doivent être ingénieurs des Ponts et Chaussées pour pouvoir tirer profit du système. La documentation automatisée va repousser ses utilisateurs tant et aussi longtemps que ceux-ci devront s'en remettre à la disponibilité, aux capacités et à l'humeur d'un expert imposé pour communiquer avec les banques de données. L'informatique juridique est vouée au rejet si le juriste/client voit son intimité brimée par une paperasserie/contrôle qui fiche son activité, ses faiblesses et ses erreurs dans une bureaucratie indiscreète²⁶.

Le concept d'accès direct et facile au système est primordial au développement de l'informatique qu'on doit considérer comme une commodité de la vie courante, et non comme un privilège ou une odyssée en l'an 2001. La complexité des procédures d'accès de D.P.S. a empêché le développement de MØDUL/DEPLOI et la nécessité d'un consultant/expert a contribué à la disparition de DATUM. Le système d'informatique juridique doit égaler en simplicité et en richesse de données la bibliothèque traditionnelle, sous peine de voir cette dernière continuer à monopoliser les voies d'accès à l'information juridique en dépit de ses handicaps évidents comme instrument de recherche²⁷.

26. Il y a peu de textes portant sur l'acceptabilité des systèmes par leurs utilisateurs ; on pourra peut-être consulter sur ce sujet : B. SCOTT ELLIS, « Human Factors Aspects of Computer-Assisted Law Searching » in *Sense & Systems in Automated Law Research*, *supra*, note 11, pp. 147-155.

27. Profitons-en pour rappeler ici la loi de Mooers : « An information retrieval system will tend *not* to be used whenever it is more painful and troublesome for a customer to have information than for him not to have it » (1960) *American Documentation*, ii.

Au plan de la richesse des banques opérationnelles, des progrès ont également été enregistrés... aux États-Unis en tout cas! FLITE et JURIS donnent accès au *United States Code* et quelque trente-cinq États opéraient en 1977 un système de repérage d'informations. En jurisprudence, LEXIS et WESTLAW s'ouvrent de vastes marchés par la seule dimension de leurs banques respectives de données. WESTLAW, par exemple, permet un repérage des résumés de toutes les décisions publiées par l'éditeur WEST depuis 1966. LEXIS s'est pour sa part spécialisé sur le riche marché de l'est (dans les États de New York et d'Ohio surtout) et par rapport au domaine lucratif de la fiscalité. Il est sûr que l'affaire est à suivre!

Par ailleurs, et ce point est peut-être plus intéressant encore, les services offerts par les Centres d'informatique juridique se diversifient au fil des ans. Aspen Systems ne limite plus ainsi ses activités à la seule recherche documentaire. Aspen crée, par exemple, des index à la carte pour les praticiens du droit impliqués dans des dossiers importants²⁸, comme il peut aider à l'implantation de systèmes ou à la publication d'ouvrages spécialisés. WESTLAW, par ailleurs, s'appuie naturellement sur l'édition traditionnelle de West Publishing, ce qui lui assure une présence sur deux plans de pénétration commerciale, une position qui ne peut que lui être bénéfique.

Parce qu'ils ont enrichi leurs banques de données et élargi la gamme de leurs services, les entreprises commerciales américaines ont aujourd'hui assuré l'avenir que la seule dimension de leur clientèle n'avait pu soutenir au départ. Toute proportion gardée, nous pouvons tirer nous-mêmes de ces réussites des leçons pour une mise en marché que nous devons aussi mener à terme chez nous, même au niveau de la clientèle privilégiée que nous avons visée à MØDUL, soit celle des fonctionnaires juristes en général, mais surtout rédacteurs de lois, légistes, voire législateurs.

1.1.3. Acquérir la diversification

Pour cette clientèle, les Américains ont fait beaucoup encore en se posant comme objectif de traiter la chaîne documentaire complète de cette unité bibliographique qu'est une loi. À ce jour, quelque trente-cinq États²⁹

28. Il s'agit du système *Instant Index*; voir H. B. FEINNEAU, « ASPEN: Organization of Litigation Files », in *Automated Law Research*, *supra*, note 11, pp. 143-145.

29. Nous tirons ici nos renseignements principalement de: L. SHULTE, « A Survey of Computerized Legislative Information System », (1979) 72 *Law Library Journal*, 99-129; R. L. CHARTRAND and J. BORTNICK, « State Legislatures and Information Technology », (1978) 11 *Law and Computer Technology*, 39-59; L. POLI, « A Legislation Information System: from Research to Use », in *Why legislators...*, *supra*, note 23; P. WEBER, « The Legislative Documents Handling System in the State of Oregon », *id.*: J.-P. ÉMARD and J. B. STAENBERG, « An Overview of Computerized Legal Information Systems — An Update », (1977) 10 *Law and Computer Technology*, 2-16.

ont informatisé au moins une partie importante des blocs d'information qu'ils doivent gérer, en particulier, aux niveaux de ce qu'ils appellent d'abord le *bill processing* (formé de sous-systèmes ou de systèmes reliés de *bill drafting*³⁰ et refonte des lois (*statutory revision*)), à celui du *bill status systems*³¹ puis à celui du repérage d'information (*statutory retrieval*). De ce nombre, les États de Washington³², d'Illinois³³, de Pennsylvanie et d'Oregon³⁴ semblent bien avoir réussi à compléter la chaîne de traitement entière depuis la rédaction du projet original jusqu'à l'impression de la loi sanctionnée. À cet effort des États américains, il faut ajouter évidemment ceux qui ont été réussis par le Congrès américain avec le système SCORPIO³⁵.

Comme on le constate lors de l'énumération précédente, l'informatique juridique des années '70 a ouvert des horizons nouveaux en facilitant l'accessibilité aux systèmes de traitement de l'information, en élargissant les banques de données déjà conçues et en fournissant le traitement des lois sur une chaîne documentaire complète. De là peut-on maintenant conclure en élaborant les principales caractéristiques que doit posséder aujourd'hui un système de traitement de l'information juridique des années '80.

1.2. La maîtrise de l'informatique

Même si nos commentaires précédents ont été élaborés à partir de données américaines, nos remarques suivantes vont s'établir sur les assises plus larges de notre expérience des systèmes de repérage canadiens et

-
30. Voir plus spécifiquement : J.V. STAPLES, « Bill Drafting and Publication Using Computers », in *Why Legislators...*, *supra*, note 23.
31. Les systèmes permettant de connaître l'état des projets de lois (*Bill Status Systems*) comptent parmi les plus anciens et les plus nombreux systèmes du monde diversifié de l'informatique juridique. Ne pouvant rapporter ici de bibliographie exhaustive, nous renvoyons par exemple le lecteur à : R. D. STEIGHNER, « Pennsylvania Legislative Data Program », in D. Johnston (ed.) *Proceedings of the Computer and the Law Conference*, 1968, Kingston, Ont., Queen's University, 1969, 104-108; « *The Commonwealth of Pennsylvania: Computerized Legislative Information Processing and Retrieval System: Application brief for IBM; Instant Legislative Information System: Florida Pioneers Data Bank for Lawmakers, RCA/Electronic Data Processing; Washington State Legislative Information System, Data Processing Application for IBM.* »
32. R. O. WHITE, « LIS: Legislative Information System for Washington », in *Automated Law Research*, *supra*, note 11, pp. 113-117.
33. R. M. HULL, « Legislative Computer Applications: the Illinois Story », (1974) 3 *Rutgers Journal of Computers and the Law*, 187-201.
34. Il s'agit ici du système OLIS; voir : P. WEBER, « The Legislative Document Handling System », in *Why Legislators...*, *supra*, note 23.
35. C. A. WOODY *et als*, « A Subject-Content Oriented Retriever for Processing Information Online (SCORPIO) », (1977) 10 *Law and Computer Technology*, 26-36 (2d quarter issue); la Chambre des Représentants n'est pas inactive non plus — voir : *The United States House of Representatives Computer-Based Bill Status System*, in *Why Legislators...*, *supra*, note 23.

européens dont nous possédons quelque connaissance. Nos conclusions surtout seront ainsi conçues en fonction de leur environnement réel, qui se dessine dans un milieu différent de celui du monde juridique américain. Notre pratique du droit s'en distingue à tous les points de vue et notre système juridique, échafaudé sur la structure du texte de loi, s'accommoderait mal des seuls processus américains où les décisions des tribunaux occupent le point de mire des raisonnements légaux.

Partant dès lors de ces bases élargies, nous avons retenu en vrac pour dessiner le système d'informatique juridique des années '80, les facteurs :

- a) Du *know-how* documentaire déjà acquis (au plan de la méthode du texte intégral, par exemple);
- b) De la somme, de la nature et du succès des systèmes présentement opérationnels et dont le bilan général est globalement positif;
- c) Des milieux sociaux et professionnels diversifiés où s'intègrent les systèmes, compte tenu qu'un praticien du droit, un professeur ou un légiste risquent d'aborder des problèmes identiques à partir des points de vue différents;
- d) Des nécessités et besoins réels des professions et du public affecté par leur exercice, en fonction des services produits par ces systèmes et de leurs coûts d'implantation et d'opération;
- e) De l'accessibilité même indirecte au plus grand nombre des citoyens, quand les projets sont supportés financièrement par des subventions gouvernementales;
- f) De l'état actuel des techniques informatiques, de la bureautique et de la télématique, et en considérant aussi les développements éventuels de tous les moyens de communication³⁶; et,
- g) Du degré de possibilité de réalisation de certaines applications sophistiquées de l'informatique, classifiées indistinctement sous l'étiquette générale d'*intelligence artificielle*.

Dès lors, j'aurais tendance à énoncer dix caractères, pas tous nouveaux d'ailleurs, qui vont former l'essence des systèmes d'informatique juridique « ordinaires » des dix prochaines années. Cinq d'entre eux se rattachent à l'architecture des systèmes³⁷, et les cinq autres portent sur leur accessibilité.

36. Un ouvrage intéressant sur ce sujet: L. BRUNEL, *Des machines et des hommes*, Québec, Québec-Science, 1978.

37. Nous employons ce terme ici dans le sens que lui entendent S. NORA et A. MINC dans leur rapport *L'information de la société*, Paris, La documentation française, 1978, p. 142.

1.2.1. La structuration en réseau

Les systèmes affectés à l'informatique juridique seront désormais *structurés en réseaux* de banques de données.

Jusqu'à ce jour, les banques de données juridiques ont presque toujours respecté un cloisonnement qu'on avait d'abord échafaudé pour empêcher les doubles traitements dus à la concurrence des pionniers zélés, des sources légales ou jurisprudentielles. Nous payons aujourd'hui le prix de la paix en termes d'isolements qui n'ont plus leurs raisons d'être. Seuls les têtus irresponsables peuvent encore croire qu'un seul centre de traitement peut gérer avec succès et profit toutes les sources de données légales, d'où qu'elles émanent.

Avant le développement des techniques de télécommunications, qui n'en est d'ailleurs qu'à faire ses premiers pas dans l'orbite des satellites spécialisés ou en suivant le fil prometteur de la fibre optique, on craignait encore que la multiplicité des centres n'engendrât l'irréremédiable anarchie des formes et l'inaccessibilité consécutive des informations traitées. La télé-informatique moderne a écarté aujourd'hui un si grand nombre des obstacles techniques et financiers qui en freinaient le développement, qu'elle en a assuré la crédibilité de l'informatique décentralisée qui dicte maintenant la règle d'exploitation normale de l'informatique nouvelle.

L'informatique en réseau est presque nécessaire pour nous, non seulement au plan inter/systèmes (entre MØDUL et DOCILIS par exemple), mais dans une perspective interne, en donnant la préférence à l'intérieur d'un macro-système, à la satellisation simple des bases plutôt qu'à la soudure figée des sources dans des systèmes monolithiques, lourds et coûteux, comme on en a érigés beaucoup jusqu'à présent.

La conception de ces systèmes modifie naturellement l'approche traditionnelle du documentaliste, qui dirige le plan d'automatisation avec la collaboration de l'informaticien. Il lui faut ici élaborer par exemple des protocoles de connexion (ou *pointeurs*) entre les bases composantes, qui ne se contenteront plus de souffler à l'utilisateur un renseignement en rubrique, mais qui l'informeront du site des données auxiliaires ou consécutives (v.g. législation subordonnée par rapport à législation principale) dont il pourrait avoir besoin, et qu'il pourra au surplus appeler immédiatement en sortie des textes dont il aura l'adresse en mémoire.

L'informatique de fichier actuelle permet de tels renvois, mais au prix de structuration de systèmes grandioses, monolithiques, rigides et coûteux. Peut-être l'époque des dinosaures tire-t-elle en somme à sa fin !

1.2.2. L'hétérarchisation des systèmes

Bien que décentralisés, nous resterons cependant, je crois, cartésiens et hiérarchiquement structurés, au moins à ce plan où nous serons toujours tentés de considérer l'une des banques/sources comme primataire par rapport aux autres.

Cette sous-banque privilégiée sera sans doute celle de la loi puisque nos systèmes juridiques considèrent la norme édictée par le législateur comme fontaine première de toutes les autres sources du droit. Contrairement au processus de la common law, notre théorie générale fait que le tribunal interprète le droit sans en principe le créer, alors que la doctrine commente la jurisprudence de son côté sans intervention actuelle sur l'application de la règle³⁸. Toute occurrence d'acte légalement significatif émane donc en dernier ressort de la loi, et de là sa prééminence naturelle par rapport aux autres sources du droit, et aux autres fichiers des macro-systèmes actuels de l'informatique juridique.

Bientôt, nous pourrons cependant réaliser, grâce à l'architecture décentralisée de nos systèmes, notre vieux rêve de pouvoir passer de la recherche par une source du droit des informations requises, à une autre source, sans pour tout cela que personne n'ait à entretenir un système gigantesque unique de toute la documentation juridique nécessaire. Tour à tour, et sans coûts exorbitants, nous pourrons rechercher la loi par la jurisprudence ou la jurisprudence par la loi, chacune de ces sous-banques jouant en son temps, son rôle de sous-banque privilégiée. Nos systèmes en réseau ne seront pas hiérarchiques, mais plutôt hétérarchiques³⁹.

1.2.3. La jonction des chaînes documentaires

C'est ainsi qu'on pourra enfin construire facilement *a posteriori*⁴⁰ une chaîne de recherche complète et unique pertinente à un problème juridique. Il n'est pas possible encore aujourd'hui de monter un dossier complet, législation et jurisprudence, sur un sujet quelconque. Tout au plus peut-on,

38. L'influence de la jurisprudence et de la doctrine peuvent se faire sentir à l'émission de la règle de droit, mais en différé seulement ; *vide* : « La machine et le droit... », *supra*, note 11, aux pages 492 et 493.

39. Un logiciel est « hétérarchique » quand chaque segment du programme global se réserve la possibilité, à tout instant, d'exploiter tout autre segment. Voir W. SKYVINGTON, *Machina Sapiens*, Paris, Seuil, 1976, p. 277.

40. Nous entendons par recherche *a posteriori*, celle qui conduit à la découverte de normes déjà existantes ; nous entendons plus loin par recherche *a priori*, celle qui conduit à l'élaboration de normes nouvelles. La première est plus coutumière au praticien qui règle un problème actuel, alors que la seconde appartient plutôt au légiste qui crée un message légal nouveau.

sauf rares exceptions, compiler des chaînes parallèles (législation ou jurisprudence) et encore ces chaînes sont-elles incomplètes parce que limitées dans le temps. De là, des silences... ! De là, la frustration du chercheur... !

Dans l'état actuel des choses, les recherches *a priori*, c'est-à-dire pour constituer des normes juridiques, ne sont pas plus faciles. Au mieux, s'aperçoit-on que certains législateurs ont automatisé la chaîne entière de la création et publication de leurs lois nouvelles. La cohérence de la chaîne de ces traitements n'entraîne pas toutefois nécessairement pour tout cela que la communication soit possible, aisément du moins, entre tous les maillons de la chaîne, qui risquent d'enfermer leurs données respectives dans des fichiers pratiquement inaccessibles. De plus, le rédacteur n'a pu pour tout cela inscrire dans un même processus la donnée jurisprudentielle peut-être nécessaire à sa rédaction ou même simuler, comme il eut pu le souhaiter, des prototypes de textes à partir desquels le législateur aurait pu faire le bon choix.

Chaînes parallèles, chaînes brisées, voilà le lot débraillé que l'informatique juridique actuelle apporte le plus souvent en dot à un chercheur surpris, dont on doit sans doute comprendre la déception quand il n'a trouvé que désordre, là où on lui avait promis structure et classement rationnel.

1.2.4. L'indexation différée

Le juriste doit par ailleurs se faire à l'idée qu'il devra continuer à maîtriser la technique des requêtes booléennes puisqu'il semble bien que personne ne va plus tenter d'assortir ses banques de données d'index hiérarchiques préstructurés, à moins que celui-ci n'existe déjà comme c'est le cas pour WESTLAW. Il incombera donc au chercheur de créer *ad hoc* sa propre transaction d'index à la carte, suivant la forme que l'on connaît maintenant... ou presque, car l'*interrogation en langage naturel* risque sans doute de devenir la règle générale de technique documentaire, suivant l'usage du texte intégral des données. Mais alors, on doit prévoir...

1.2.5. Des logiciels traducteurs

Ce n'est sans doute qu'au terme de la seconde partie de cet exposé que l'importance véritable du point que nous soulevons ici, apparaîtra dans les justes dimensions que ce problème revêt en fait, puisque nous demanderons alors au système de repérage intelligent idéal que nous dessinerons là encore, de comprendre⁴¹ les interventions de son interlocuteur, et non simplement

41. Plus loin, nous suivrons par l'intermédiaire de W. SKYVINGTON (*Machina sapiens, supra*, note 39), la classification des systèmes intelligents conçue par Pierre Schaeffer, où l'on y distingue les systèmes qui s'arrêtent aux *mots ouïs* ou aux *mots écoutés*, par rapport à ceux qui perçoivent des *mots entendus*, voire des *mots compris* (p. 175); voir, *infra*, note 43.

de produire mécaniquement des sorties d'information, parce qu'il aura perçu (ou ouï) des signes-repères dans un assemblage de super-signes, comme le font nos systèmes de repérage actuels, y compris ceux qui semblent saisir le langage naturel⁴².

Sans atteindre pour le moment la qualité de l'analyse sémantique, mais en dépassant néanmoins l'humble stade du faux dialogue produit par des systèmes ingénieux, nous croyons que le système de repérage des années '80 devrait pouvoir au moins « écouter » des ensembles linguistiques et même en analyser la structure grammaticale par des modèles Chomskiens⁴³. Irons-nous plus loin ? La ligne reste bien grise entre le système automate et celui qui peut réaliser une analyse conceptuelle⁴⁴.

Faute d'utilité pratique à court terme puisque ce débat sera repris plus loin, ne tranchons pas ici la question sans avoir au préalable poussé notre étude plus avant sur les systèmes dits intelligents. À tout événement, le logiciel traducteur que nous envisageons maintenant devrait être capable d'« écouter » des interrogations en langage naturel quand ce ne serait, avouons-le, que pour ouvrir la porte à la flexibilité d'analyses conceptuelles ou sémantiques comme nous souhaiterions bien que nos systèmes puissent le faire.

1.2.6. Des systèmes interactifs

Cette norme est maintenant si généralement respectée par presque tous les responsables de systèmes de repérage d'information que nous la rapportons ici presque uniquement par acquis de conscience. Il reste encore

42. Les faux dialogues poursuivis par les systèmes interactifs, mais non vraiment conversationnels, s'illustrent facilement par des logiciels comme STAIRS. Les systèmes *Eliza* ou *Paranoia* fournissent cependant de plus justes exemples encore de ces conversations-bidons où l'une des parties ne répond pas du tout à son interlocuteur, ses interventions étant tout simplement mécaniquement produites par la reconnaissance de mots-repères formellement reconnus dans les messages livrés par l'autre partie (SKYVINGTON, *supra*, note 39, pp. 176 à 185).

Certains systèmes qui sont présentement opérationnels, se targuent de maîtriser l'interrogation en langage naturel. Il faut cependant n'enregistrer ici ces renseignements qu'avec prudence et ne pas s'imaginer surtout que ces systèmes dialoguent. Ils ne sont en fait ainsi opérationnels qu'en raison d'un ingénieux procédé qui réduit les interventions du requérant en un énoncé quasi booléen ordinaire, une fois que les énoncés traités par l'ordinateur, ont été dépouillés de leurs mots vides, et que les autres termes de la requête, ont été évalués de quelque façon par le système (voir: J. BING, « Legal Information Retrieval Systems: The Need For And The Design Of Extremely Simple Retrieval Strategies », (1978) 1 *Computer/Law Journal*, 379-399).

43. Le système qui « écoute », peut procéder à des analyses syntaxiques ; celui qui « comprend » peut réaliser une analyse sémantique. Entre les deux, le système qui « entend » peut procéder à des analyses conceptuelles (*Id.*, p. 175).

44. Voir nos remarques en note précédente.

aujourd'hui naturellement le prestigieux système FLITE qui opère par lot en mode différé, et en faisant appel à un juriste/interprète de requêtes, comme c'était le cas auparavant à DATUM⁴⁵. Ce cas est cependant exceptionnel et on notera au surplus que son propriétaire, l'Aviation américaine, est *en même temps* relié à JURIS, qui, lui, permet l'interaction entre le système et son utilisateur. Deux systèmes pour une même institution ! Qui dit mieux ! Mais qui a les moyens de se payer ainsi les économies qu'apporte à l'Aviation américaine le système fondamental en différé dans de telles conditions ? Quoi qu'il en soit, il faut impliquer le chercheur dans le processus de traitement de sa propre requête. Il en est le créateur et lui seul peut la conduire efficacement à son point de résolution. Le traitement par lots alourdit de façon intolérable sa démarche.

1.2.7. Des systèmes accessibles

Comme nous l'avons dit déjà, l'informatique juridique n'a pas d'avenir si elle ne s'inscrit pas au plus vite à part entière dans le mouvement télématique actuel, où l'informatique passe de son état primaire de nouveauté exotique à celui plus normal d'outil de travail ordinaire conçu pour le lot général de la population. L'ordinateur, en d'autres termes, prend aujourd'hui sa nature essentielle de *medium de masse*, dont il avait été jusqu'ici éloigné pour des motifs purement techniques, en raison du coût et de la taille des ordinateurs de grande puissance qu'on a fabriqués jusqu'à maintenant, et à cause également de la complexité des langages dont on avait

45. Cet interprète nous semble créateur d'interférences parce qu'il constitue un « écran » entre le système et son utilisateur. Il faut noter ici toutefois que la clientèle de FLITE n'est pas composée exclusivement de juristes, et qu'on doit comprendre que ces clients ont effectivement besoin d'aide pour formuler leurs problèmes. L'expérience enregistrée à ce sujet par le *Los Angeles County Law Library* à partir de l'exploitation de WESTLAW paraît concluante (voir : M.A.R. ANDERSON, « Automation in the Law School Library », (1980) 11 *Law Librarian*, 11-14, et nos remarques, *infra*, en note 49). Il faut toutefois consulter sur ce sujet l'exposé le plus complet à ce propos in : E. MACKAAY, « Reflections on the First National Conference Automated Law Research », (1974) 3 *Rutgers Journal of Computers and Law*, 310-327 (320-322). Dans une étude qu'il a remise dernièrement à SOQUIJ, notre ami et collègue Ejan Mackaay semble se ranger avec ceux qui prônent les avantages du système « libre » ; il affirme ainsi que : « Pour mettre toutes les chances de son côté, un... [système de documentation juridique automatisée] devrait permettre l'accès direct par terminal. » (E. MACKAAY, *La documentation juridique au Québec au début des années 80 : les résultats d'un sondage*, document dactylographié remis à SOQUIJ à Montréal en août 1980). On constate cependant en revanche que l'intervention d'un spécialiste qui puisse opérer le terminal et aider à formuler les requêtes, reste appréciée par beaucoup d'utilisateurs, surtout dans les cas où les interrogations se font à partir d'un terminal « public » (cf : F. M. GREGURAS et L.L. CARLILE, « Report of the Public Terminal Study », (1978) 1 *Computer/Law Journal*, 255-357, à la page 275) ou dans le cas où le requérant est un juriste plus âgé (cf : R. M. MCGONIGAL, *supra*, note 19).

jusqu'ici essentiellement besoin. Ce n'est pas tout le monde qui peut, ou désire, parler en Assembleur !

Avec l'apparition des microprocesseurs⁴⁶ (*chips*), et de l'ordinateur individuel⁴⁷, et, bientôt, espérons-le, de langages de plus en plus flexibles⁴⁸, tout le monde de l'informatique se modifie en profondeur, la clientèle de cette discipline risquant de se diversifier considérablement. Les étudiants de Québec qui seront juristes en 1990, connaîtront le micro-ordinateur et souhaiteront travailler seuls à un terminal avec une insistance irrésistible, là où les plus âgés doivent encore vaincre une certaine résistance face à la machine⁴⁹. Ceux qui auront fait l'expérience pratique de la télévision bidirectionnelle⁵⁰, ne vont pas comprendre qu'on s'interpose entre leur processus de recherche et la banque des données qui peut leur fournir l'information recherchée.

Il nous faut donc, comme le disait McLuhan, cesser de voir l'avenir par le rétroviseur du passé⁵¹, et concevoir un système désormais en fonction des progrès éventuels inéluctables des logiciels d'intelligence artificielle, des

46. C'est semble-t-il à « Ted » Hoff, un ingénieur électricien, que nous devons le *chip*, lancé sur le marché en 1971. Sur une surface de 1/6 pouce × 1/8, le microprocesseur travaille comme 2250 transistors et complète un travail de calcul équivalant à un ordinateur de deuxième génération dont le prix s'élevait à quelque \$30 000 au début des années '60. Voir: «The Computer Society», *Time*, le 28 février 1978, pp. 42 et *sq.*, ainsi que «Machines That Think», *Newsweek*, le 30 juin 1980, pp. 50 et *sq.*

47. Il existe déjà une littérature si abondante sur le micro-ordinateur qu'il serait impossible de fournir ici une bibliographie complète sur ce sujet. Le nouveau venu à cette discipline pourrait peut-être consulter toutefois à ce sujet: Y. LECLERC, *L'ordinateur individuel*, Montréal, L'étincelle, 1979.

48. Pour l'instant, le langage BASIC paraît avoir pris une avance sur tous les autres; son plus proche concurrent serait PASCAL.

49. Lorsque le Ministère canadien de la Justice a réalisé l'opération *Complex* entre juin 1971 et avril 1972, les avocats canadiens ont déclaré préférer un système où un juriste/interface ferait la recherche à l'ordinateur en leur nom (voir: *Opération Complex*, Ottawa, Ministère de la Justice, 1972, p. 26). Or, on sait maintenant par DATUM, que cette addition au système général ne constitue pas une réponse suffisante en soi à l'attitude réticente de la clientèle éventuelle. Voir nos remarques, *supra*, note 45.

50. En télévision bidirectionnelle, le récepteur classique devient terminal en accouplant son action à celle de l'appareil de téléphone ordinaire. La nouvelle télévision permet à l'ancien téléspectateur physiquement *passif*, de se *bâtir* maintenant une carte de programmes à son choix, et même d'intervenir directement dans certaines émissions participatives. Les principaux systèmes qui sont présentement opérationnels et en cours de tests dans certaines régions, sont QUBE (Columbus, Ohio), ANTIOPE (France), Prestel (Post Office britannique) et TELIDON (voir: D. WRIGHT, « Une technologie de pointe à la recherche d'un marché », (1979) 6 *En quête*, 2, (no d'hiver).

La même technologie qui peut transmettre les petites annonces de l'immobilier, peut téléistribuer des textes juridiques!

51. M. McLuhan, *The Medium is the Message*, New York, Bantam Books, 1967.

développements des supports de quincaillerie de l'informatique fondamentale et par rapport également à notre nouvelle vie dans la cité câblée⁵² d'aujourd'hui. Un récepteur de télévision ordinaire peut constituer un terminal opérationnel valable.

1.2.8. Des systèmes de saine ergonomie documentaire

C'est en réalité pour des motifs de rentabilité que nous devons insister pour que nos systèmes d'informatique juridique soient pleinement accessibles à ses utilisateurs. Autrement, suivant en cela peut-être la Loi de Mooers⁵³, ces machines à renseigner ne seront pas mises à profit et rateront par conséquent leurs objectifs.

Il n'est pas fantaisiste d'intégrer à une banque de données, index internes, table des matières et plans d'ensemble qui aident l'utilisateur à se situer par rapport à l'appareil informatif global, dont il forme d'ailleurs lui-même un élément essentiel, puisqu'il en constitue l'indispensable récepteur.

Au chapitre de l'ergonomie documentaire et des techniques d'accès aux systèmes, nous aurions beaucoup à retenir du système d'enseignement avec aide d'ordinateur PLATON⁵⁴, qui rend disponible à ses utilisateurs une première leçon dont le but consiste à enseigner l'usage de PLATON lui-même. Pourquoi ne retiendrait-on pas également l'idée d'incorporer à nos systèmes des fonctions de « dépannage » comme la touche-fonction HELP⁵⁵

52. Voir: L. BRUNEL, *supra*, note 36, pp. 61-72; il y a dix ans déjà, on parlait beaucoup de la cité câblée dans *Univers sans distance: rapport sur les télécommunications au Canada*, Ottawa, Information Canada, 1971, chapitre XV.

53. *Supra*, note 27.

54. PLATON (ou PLATO) est un système multimédia d'aide à l'enseignement sur support informatique. On en doit la conception surtout au Dr Donald L. Bitzer du Computer-Based Education Research Laboratory de l'Université d'Illinois (Urbana-Champaign, Ill.), qui amorça ses travaux de recherche sur PLATON dès 1960. Opérationnel sur les ordinateurs de Control Data Corporation, qui a supporté financièrement le projet surtout entre 1967 et 1972, PLATON fonctionne en langage TUTOR, conçu originalement par Paul Tenezar.

Une riche bibliographie a été publiée sur PLATON, mais nous ne soulignerons ici que: S. G. SMITH et B. A. SHERWOOD, « Education Uses of PLATO Computer System », (1976) 192 *Science*, 344-352; P. B. MAGGS and T. D. MORGAN, « Computerized Legal Education », (1975) 27 *Journal of Legal Education*, 138-156; H. G. HENN et R. C. PLATT, « Computer-Assisted Law Instruction: Clinical Education's Bionic Sibling », (1977) 28 *Journal of Legal Education*, 423-436; Robert J. MANRO et D. NOAH, « PLATO, EDUCOM, and Legal Education », (1980) 30 *Journal of Legal Education*, 582-591.

55. Une fonction n'est pas toujours actualisée sur le clavier du terminal par une touche spécifique, mais c'est le cas pour HELP à PLATON. Cette fonction fait générer une séquence d'instructions conçues pour aider l'utilisateur à retrouver la procédure correcte d'opération qu'il aurait pu oublier, ou encore, de lui « souffler » une bonne réponse qu'il n'aurait pas réussi à trouver (voir: *Control Data PLATO Terminal User's Guide; Control*

de PLATON, qui pourrait aider à corriger les démarches fautives des novices de l'informatique juridique, ou à relancer la recherche d'un expert qui souhaiterait s'inspirer d'instruments de recherche auxiliaires comme le sont les vocabulaires⁵⁶, listes de fréquence ou listes grammaticales ou synonymiques⁵⁷ dont nous connaissons bien l'utilité déjà?

Il est aujourd'hui démontré en effet que l'intégration totale d'un thésaurus à un système de repérage d'informations se révèle si coûteux et si peu utile qu'on ne peut généralement justifier une telle opération⁵⁸ mais rien n'empêche que ces listes ne constituent des bases satellites d'accès optionnel, surtout dans l'optique globale des systèmes en réseau que nous avons évoqués déjà. Puisque les raisons les plus convaincantes qui militent contre l'intégration des thésaurus, sont le plus souvent économiques, et pour des motifs surtout techniques (v.g. coûts de maintenance en mémoire), peut-être pourrait-on même imaginer ici des formules de rechange elles-mêmes techniques, mettant à profit l'institution de mini-systèmes parallèles centrés sur l'exploitation de micro ou même de pico-ordinateurs⁵⁹.

Data PLATO System Overview; W. SCHWAB, *Le guide du médiatiseur : introduction au langage PLATON*, Québec, Université du Québec, 1979.

56. Ces vocabulaires, dictionnaires, listes de fréquence et listes inversées constituent le plus souvent des produits *standard* des progiciels les plus connus (v.g. STAIRS). Leur consultation reste toujours possible sur papier hors du mode opérationnel du système lui-même.
57. Les listes et thésaurus doivent être composés de toutes pièces par les artisans des systèmes concernés. À MØDUL/Deploi, nous devons ainsi des listes « g » et « s » de grande qualité à l'action concertée de nos collègues Michael Mephram, du département de linguistique de l'Université Laval, et Jeanne Leclerc-Houde, qui fut longtemps reliée au développement de Deploi.
Tous les auteurs de listes francophones doivent beaucoup à Lucien Mehl et son équipe du CEDIJ; voir nos remarques à « La machine et le droit... », *supra*, note 11.
58. On doit aux artisans de DATUM la confection du meilleur thésaurus, bilingue au surcroît!, que l'informatique juridique ait jusqu'ici aidé à produire (à ce sujet, voir: le numéro spécial de la *Revue juridique Thémis*, de 1976; E. MACKAAY, « The Creation of a Bi-Lingual Thesaurus for a Full Text Retrieval System (DATUM) », (1973) 6 *Law and Computer Technology*, 2-12.
Pour des raisons qui restent obscures, le thésaurus est resté presque inutilisé par la clientèle et les consultants de DATUM qui semblaient pouvoir s'en passer aisément. Un tel phénomène s'observe ailleurs (v.g. MØDUL/Deploi), si bien qu'il faudrait en conclure que les thésaurus sont utiles, mais non nécessaires, à l'exploitation d'un système d'informatique juridique à base de texte intégral, lorsqu'opéré par des juristes.
59. La micro-informatique que nous connaissons actuellement, est encore consacrée presque entièrement à la gestion d'entreprises ou conçue en fonction d'exploitations spécialisées et parfois presque frivoles. Rien n'empêche qu'elle puisse évoluer vers des applications alphanumériques importantes (v.g. gérance de thésaurus) comme ce fut le cas pour la grande informatique traditionnelle. Enfin, on pourrait peut-être utiliser un jour des pico-ordinateurs (v.g. ordinateur à un ou quelques micro-processeurs, tels qu'illustrés par les jeux Mattel ou, plus sérieusement, par les calechettes sophistiquées de Sharp, Hewlett-Packard ou Texas Instruments).

1.2.9. Des systèmes sympathiques

Retenant encore une fois le système PLATON⁶⁰ comme exemple principal, il nous faut réaliser en effet que nos systèmes doivent, pour être acceptés, obéir aux lois normales du marché moderne de la communication, et se présenter dans des habits qui attirent le consommateur, et se comporter suivant des schémas qui correspondent au préacquis de la civilisation des media de masse électroniques.

Notre conditionnement fut-il en réalité subliminal, nous sommes habitués à un écran de télévision qui apporte son et image dans le style décontracté que maîtrise fort bien la publicité⁶¹ qu'on nous présente chaque jour à fortes doses par nos téléviseurs individuels. Tout message, médiatisé par le support électronique, doit obéir aux règles et normes de communication de ce médium, sous peine de se présenter à son récepteur dans une forme inacceptable pour ce dernier, qui lui opposera dès lors un refus global. Comme le dit le juriste communicateur Jean Cloutier : « ... chaque médium a un langage qui lui est propre »⁶².

La position de l'informaticien du droit reste sur ce plan fort difficile à situer dans un juste milieu efficace, puisqu'il lui faut ne pas désarçonner son utilisateur par une présentation de l'information si peu familière que son utilisateur n'accorde pas de crédibilité à l'information reçue et, en même temps, éviter de concevoir des systèmes calquant de si près l'information imprimée, que ces derniers ne convaincront pas plus la communauté juridique que le premier. Même l'utilisateur le plus conservateur s'attend à juste titre à ce que le système informatique lui apporte quelque chose de plus qu'il ne trouve pas dans ses livres traditionnels, et il ne faut pas tromper cette attente.

Or, l'élément nouveau que génère l'information électronique par rapport à l'imprimé, réside justement dans le *mouvement* que permet ce médium. De

Ces pico-ordinateurs pourraient entreposer en mémoire et traiter des listes synonymiques limitées pour des applications spécialisées (v.g. termes reliés, anglais-français-allemands en droit des corporations).

60. PLATON constitue l'un des rares systèmes informatiques de haute diffusion que l'on puisse réellement considérer comme un *medium* d'information. Le jargon informatique américain parle ici de *friendly system*.
61. Au risque de choquer, je crois devoir affirmer que les juristes communiqueraient mieux s'ils s'inspiraient des techniques publicitaires. Marshall McLuhan a visé juste en faisant remarquer que la publicité reflétait mieux peut-être que toute autre forme d'expression, les préoccupations, les mœurs et les modes de pensée contemporains, en caricaturant avec précision les archétypes sociaux pertinents à chaque époque; voir : *The Mechanical Bride*, Boston, Beacon Press, 1951. Dans son langage imagé, McLuhan dit encore : « Bless Madison Avenue for restoring the Magical Art of the Cavemen to Suburbia », *Counterblast*, London, Rapp Whiting, 1970.
62. J. CLOUTIER, *L'ère d'Emerec*, 2^e éd., Montréal, P.U.M., 1975.

là, avons-nous suggéré déjà l'accès à des systèmes complets, hétérarchiques en réseaux communicants.

Plus qu'un système, un véritable médium de communication, PLATON peut nous servir ici de modèle à suivre. Il ajoute en effet à ses capacités d'éditeur traditionnel des possibilités de graphisme illustratif qui retiennent l'attention de son utilisateur et le rendent « amical » dans son dialogue interactif. Même son écran tactile pourrait nous être utile. La qualité de communication d'un système informatique n'est pas un luxe capricieux !

1.2.10. Des systèmes économiques

Les systèmes informatiques des années '80 devront être par ailleurs économiques. Le problème semble banal, presque un lieu commun, mais en vérité il n'est pas facile d'établir le coût mensuel véritable d'un système informatique. Plusieurs facteurs entrent en ligne de compte. On doit ainsi considérer non seulement le prix de l'abonnement au système, mais encore celui de la location du terminal et de son entretien, ainsi que le montant du traitement de tous ceux qui graviteront autour de la maintenance et de l'exploitation du produit informatique concerné. Dans le cas d'un système d'enseignement avec l'aide de l'ordinateur analogue à PLATON⁶³, il faut compter au départ, outre le prix de l'abonnement, environ \$500 supplémentaires pour la location du terminal, en plus d'autres frais pour le MODEM, les télécommunicateurs, etc. Déjà effrayés par ces « frais fixes », les administrateurs de nos maisons d'enseignement froncent encore les sourcils réalisant qu'un professeur peut devoir consacrer jusqu'à cent heures de travail à la préparation de deux heures de cours en mode informatisé⁶⁴. En pleine période d'austérité budgétaire, on sera sans doute facilement porté à considérer l'enseignement avec l'aide de l'ordinateur comme un luxe ! Pourrait-il en être ainsi pour l'informatique juridique ? Voyons quelques exemples tirés du modèle américain.

À l'heure actuelle, il en coûte environ \$1 500 par mois à une Faculté de Droit américaine pour être reliée au système LEXIS⁶⁵. Il s'agit cependant d'un « prix d'ami », car l'abonnement mensuel peut atteindre la somme de

63. Nous ne prétendons ni ne voulons présenter ici les coûts d'opération de PLATON, que nous ne citons qu'à titre d'illustration. Les chiffres que nous rapporterons se veulent volontairement approximatifs, et ne représentent pas les tarifs du système PLATON.

64. Nous prions encore une fois le lecteur de lire ces lignes avec prudence. Les chiffres que nous rapportons ici peuvent varier suivant la difficulté de la leçon traitée ou du degré de sophistication qu'on apportera à sa présentation ; néanmoins, voir : J. A. SPROWL, « Automating the Reasoning Process... », *supra*, p. 8 (note 8) et p. 61 (note 59).

65. Voir à cet effet : DEE et KESSLER, *supra*, note 19, p. 168 ; GREGURAS, *id.*, p. 359 ; J. SPROWL, *id.* p. 187 ; L. SCHULTE, *supra*, note 11, p. 129 ; M. ANDERSON, *supra*, note 63, p. 13 ; R.M. MCGONIGAL, *supra*, note 19 ; F.M. GREGURAS et L.L. CARLILE, *supra*, note 45.

\$2 500 pour un client ordinaire. Suivant le mode choisi par le client de LEXIS, le taux/horaire varie entre \$75 et \$130/heure, avec une moyenne générale apparente d'environ \$100⁶⁶. WESTLAW, de son côté semble coûter un prix à peu près identique, son tarif de lancement ayant été établi le 23 septembre 1975 au montant de \$1 200/mois, auquel il fallait ajouter \$2.50 par requête formulée entre huit et vingt heures⁶⁷.

À partir de ces quelques chiffres, nous ne devons pas nous demander si ces prix sont élevés en eux-mêmes, mais plutôt s'ils sont trop élevés par rapport aux ressources que la clientèle éventuelle de nos systèmes est prête à affecter à l'informatique. Au Canada, à tout le moins, une telle réflexion s'impose d'autant plus que ni *Case Law Research* d'Edmonton⁶⁸, ni DATUM de Montréal⁶⁹, ni même en fait Q/L d'Ottawa, n'ont réussi jusqu'ici à survivre par la seule mise en marché du commerce de la documentation juridique automatisée. La clientèle de ces entreprises a-t-elle été rebutée de l'utilisation de ces services surtout à cause de leurs prix ?

Rien n'indique qu'il en ait été véritablement ainsi. Les prix chargés par Q/L et DATUM n'ont jamais été excessifs. Pendant un bon moment, DATUM/SEDOJ offrait ses recherches pour un prix fixe de quelque \$15 à \$25 soit une cinquantaine de dollars en 1980⁷⁰. Un bon nombre de praticiens se sont prévalus de cette offre, si bien que le service DATUM s'était établi une clientèle assez stable, mais malheureusement restreinte étant donné les dimensions du marché; à sa dernière année d'opération, en 1978-79, le

66. *Idem.*

67. Voir: DEE and KESSLER, *supra*, note 19, p. 168; GREGURAS, *supra*, note 19, p. 359, R.M. MCGONIGAL, *supra*, note 19; F.M. GREGURAS et L.L. CARLILE, *supra*, note 45. Ajoutons au surplus que BOC Datasolve Ltd de Grande-Bretagne charge à sa clientèle un tarif de quelque 40 £/heure pour l'utilisation du système Eurolex en opération depuis le 29 janvier 1980. Les coûts de l'abonnement s'élèvent à £/1 000 (voir: (1980) 11 *Law Librarian* 5).

68. *Case Law Research* représente la première tentative de commercialisation de documentation juridique au Canada; cette corporation était dirigée par Keith Latta: K. LATTA, « Information Retrieval: an Automated System for Case Law », (1967) 10 *The Canadian Bar Journal*, 110.

69. Comme nous l'avons dit déjà, une résolution du Conseil d'administration de SOQUIJ, en date du 3 avril 1979, a mis fin à l'exploitation commerciale de DATUM.

70. D'après ce que déclarait le professeur Hugh Lawford à la Conférence de l'American Bar Association tenue à Atlanta en 1972, QUIC/LAW chargeait alors \$1.00 pour avoir accès au système, et un autre dollar pour toute transaction effectuée sur un secteur quelconque de la banque de données. Cette recherche ne coûtait rien toutefois, si elle prenait plus d'une minute à parvenir à l'auteur de la requête. Dans le cas de DATUM, il faut souligner ici que le montant d'environ \$25 dont on parle ici, incluait l'intervention de l'analyste chargé de formuler la requête du client concerné (voir: E. MACKAAY, « Reflections on the First National Conference on Automated Law Research », (1974) 3 *Rutgers Journal of Computer and the Law*, 310-327 à 319.

Service DATUM avait traité 874 requêtes par rapport à 752 l'année précédente et 630 en 1976-77⁷¹.

Le manque de sensibilisation des juristes face aux avantages que procure l'informatique juridique, l'obsolescence technologique et une mise en marché défectueuse⁷² ont empêché la réussite commerciale de l'informatique juridique au Canada. Les prix n'ont joué qu'un rôle mineur. L'étude Compulex du ministère canadien de la Justice démontrait d'ailleurs en 1972 que les études légales étaient prêtes à consacrer jusqu'à \$2 500 par mois pour accéder à un service de documentation automatisée. Suivant les résultats découlant de cette enquête, un cabinet canadien de taille moyenne, réunissant de 4 à 9 avocats, aurait alors consacré jusqu'à \$500 par mois pour un tel service⁷³. Reportés en dollars de 1980, les taux d'abonnement exigés par LEXIS ou WESTLAW ne se révèlent donc pas exagérés et on peut supposer qu'une étude légale canadienne de taille moyenne pourrait en principe supporter des coûts mensuels de quelque mille dollars pour un tel service⁷⁴. Il n'en reste pas moins cependant que la réalité canadienne actuelle vient contredire ces principes, et qu'il en sera probablement ainsi tant et aussi longtemps que la preuve de la nécessité de l'informatique juridique n'aura pas été établie à la satisfaction du monde des juristes.

Or, une telle démonstration ne peut être construite de façon concluante que par le message des faits, par l'utilisation intensive d'un groupe d'abonnés fidèles, qu'on doit recruter dans un milieu qui puisse se payer les frais de l'expérience ou qui ne peut échapper au phénomène informatique. Il n'y a donc ainsi que l'administration publique qui puisse supporter le lancement véritable de l'informatique juridique, parce que la taille de son contentieux

71. Les responsables de DATUM estimaient en 1978 à environ 30% le taux de retour de sa clientèle; in: *Quelques éléments de réflexion quant aux orientations de SOQUIJ...*, document dactylographié émanant de SOQUIJ, en date du 1^{er} mai 1978. Nous tenons ces chiffres du *Rapport annuel de SOQUIJ*, 1977-78, et de celui de 1978-79, *infra*, note 74, p. 21. On remarquera cependant que DATUM répondait à quelque 350 requêtes par mois à l'automne de 1972. « Tout nouveau, tout beau », dit le vieux dicton ! (E. MACKAY, note précédente, p. 324).

72. DATUM est arrivé sur un marché non préparé à recevoir ses avantages et retombées bénéfiques. La situation pourrait changer avec le temps, mais son bassin de clientèle demeure restreint.

73. Voir: *Opération Compulex*, *supra*, note 49, p. 25.

74. Il ne faut pas surestimer l'importance des grandes études légales. Les responsables de DATUM considéraient en 1978 que leur client/type était « un avocat de Montréal ou Québec, pratiquant seul ou dans un bureau de moins de 10 avocats » (cf: *Quelques éléments...*, *supra*, note 71). En 1977-78, DATUM rendait des services à des « petits bureaux » (1 à 3 avocats) dans une proportion de 43.57% par rapport à l'ensemble de sa clientèle, et à « des études moyennes » (4 à 10 avocats) pour 22.44%. (cf: SOQUIJ, *Rapport annuel 1977-78*, Montréal, SOQUIJ, 1978, p. 26).

global peut faire générer une preuve convaincante, et que l'importance de ses moyens financiers permet une telle implantation.

Dans certains cas enfin, celui du Législateur en particulier, l'informatisation est inévitable en raison du volume des données traitées par ces organismes, de la rapidité d'exécution qu'on exige de leur action, et des critères de fiabilité et de sécurité auxquels doivent répondre ces données pour être utiles à la communauté juridique. En fait, la preuve de rentabilité de l'informatique juridique existe déjà au profit du Législateur sous tous les points de vue.

Au plan des économies possibles, Chartrand et Bortnick rapportent ainsi pour l'État d'Illinois des économies annuelles de quelque \$260 000 sur ses travaux de constitution de projets de loi (*bill drafting*) en ayant éliminé 12 années/hommes en opération de dactylographie et d'années de lecture d'épreuves⁷⁵. D'autres États, New York⁷⁶ et Oregon⁷⁷ par exemple, prétendent aussi à des économies substantielles provenant de l'automatisation de leur processus législatif. Rappelons d'ailleurs que presque tous les législateurs américains ont aujourd'hui recours à l'informatique de quelque façon, trente-cinq au moins opérant des systèmes reliés au *bill processing* et autant tirant parti de systèmes de repérage d'informations⁷⁸.

Élargissant encore nos perspectives sur ce dernier point pour revenir au monde commercial, remarquons enfin que les observations pessimistes d'Opération Compulex à propos d'OBAR⁷⁹ n'ont heureusement pas été confirmées par les faits, puisque LEXIS, le successeur d'OBAR a franchi aujourd'hui le cap des deux cents abonnements⁸⁰ et proclame répondre à vingt mille (20 000) requêtes par jour⁸¹. Pour sa part, SCORPIO, un système

75. CHARTRAND et BORTNICK, « State Legislatures... », *supra*, note 29, p. 56.

76. *Ibid.*; aussi, ÉMARD et BORTNICK, *supra*, note 29, p. 12.

77. Voir la publicité de la société IBM concernant OLIS. On prétend que le législateur de l'État d'Oregon aurait économisé quelque \$300 000 par le seul biais de la photo-composition.

78. CHARTRAND et BORTNICK, *supra*, note 29, *passim*; mais en particulier les tableaux des pages 46 et 51; aussi, L. SCHULTE, *supra*, note 29, *passim*, mais sans oublier l'appendice, pp. 118-129.

79. Plusieurs textes ont été publiés sur OBAR; par exemple, R. J. ASMAN, « OBAR: Ohio State Bar Automated Research », in *Automated Law Research*, *supra*, note 19, pp. 43-47. Nous ne ferons pas ici l'histoire d'OBAR qui débute en 1967, nous contentant de souligner que le système, opéré originalement par Ohio State Bar Association Automated Research Corporation, a été repris en 1969 par *Mead Data Central* de Dayton, Ohio, qui continue aujourd'hui l'exploitation du système, devenu maintenant LEXIS. En 1972, le groupe de recherche de l'opération Compulex, remarquant que seulement huit terminaux auraient été installés chez des clients après six années d'efforts, semblait croire qu'une telle situation allait se continuer (*Opération Compulex*, *supra*, note 49, p. 21).

80. Cf: J. A. SPROWL, « Computer-Assisted Legal Research... », *supra*, note 19, p. 188.

81. Ainsi le proclame la publicité de *Mead Data Central*, parue dans le numéro de mai 1980 de l'*American Bar Association Journal* (p. 523).

opéré par des agences gouvernementales et traitant des informations législatives a vu son taux de consultation augmenter de 1000% entre septembre 1974 et septembre 1976, et prétend que chacun de ses deux cent cinquante terminaux véhiculerait quelque cinq requêtes par jour⁸².

Il faut naturellement se garder ici de transposition de chiffres bruts du milieu américain à nos environnements respectifs, les conditions propres à chacun variant trop considérablement d'une société à l'autre. Les ressources financières et professionnelles conditionnent la mise en marché du système d'informatique juridique au niveau le plus considérable. De l'expérience américaine, il n'en ressort pas moins quelques leçons qu'il serait souhaitable de retenir pour notre profit. Ainsi, il faut remarquer :

- a) d'abord, que l'explosion de l'information légale n'est pas confinée au seul ordre juridique américain et que la réponse informatique à l'entropie documentaire peut bien desservir dans son état actuel quiconque veut y recourir, qu'il soit chimiste ou juriste, nord-américain ou européen ; et ensuite,
- b) que le juriste, comme tout autre professionnel, ne boudera pas continuellement l'outil technologique, si on lui remet les clés d'un système dont il puisse contrôler, lui-même autant que possible, et aussi facilement que possible, le levier de commande et le contenu de sortie.

Lorsque nos systèmes actuels répondront aux dix critères que je viens de rapporter, il me semble que tout ou presque se trouvera alors en place pour construire la preuve convaincante que l'informatique juridique n'est pas simplement utile, mais indispensable dans notre monde d'aujourd'hui. Il serait de fait bien dommage que nous interrompions nos efforts dans ce domaine car, même dans l'état actuel de sa technologie documentaire, notre informatique juridique actuelle fait la démonstration de son efficacité et de sa rentabilité lorsqu'on s'en sert en respectant ses limites, soit en l'affectant à des tâches de repérages quantitatifs ou au plan de la précision la plus juste, mais aussi la plus strictement objective, pour rechercher par exemple un mot bien identifié.

Au delà de ces tâches, les faiblesses de l'informatique deviennent évidemment apparentes. L'ordinateur ne peut de lui-même ni généraliser ni abstraire. S'il peut trier inlassablement, il ne peut classifier ni qualifier facilement. La machine, dit-on, n'est pas intelligente. Elle reconnaît les mots, mais n'identifie ni les notions ni les concepts. Or, le juriste consacre une bonne partie de son activité professionnelle à qualifier, analyser ou juger.

82. Cf. C. WOODY *et als*, « A Subject-Content — ... », *supra*, note 35, p. 31-32.

Pour véritablement répondre aux besoins du juriste, faudrait-il dès lors concevoir des systèmes intelligents, des systèmes qui raisonnent? C'est l'opinion qu'ont émise certains auteurs, et le défi qu'ont tenté de relever certains juristes.

Dans les pages qui suivent, nous tenterons d'analyser d'abord les exposés des premiers, pour étudier ensuite les systèmes des seconds. À partir de ces données, nous amorcerons encore une fois la nomenclature des caractères dont nous souhaiterions voir nantis les systèmes de traitement de l'information intelligente qui représentent peut-être nos systèmes opérationnels dans ce domaine pour la décennie qui aboutira à l'an 2000.

2. Informatiser l'intelligence: le processus d'apprentissage

Le temps semble bien venu en effet de remettre l'informatique juridique en question, ce que nous n'avons pas vraiment fait encore jusqu'à présent, ayant célébré les vertus de la recherche documentaire automatisée, plutôt que celles de la véritable informatique juridique, celle qui conduit à l'information proprement dite plus qu'au support qui la rapporte. Or, la documentation automatisée a soulevé des critiques souvent justifiées au cours des dernières années et ces commentaires dépassaient souvent la technique même du repérage documentaire pour atteindre les racines mêmes du phénomène juridique.

Dans le rapport qu'il adressait en 1972 au ministre canadien des Communications, le professeur Philip Slayton déclarait, en conclusion de son étude sur la recherche documentaire électronique dans les sciences juridiques⁸³, ses craintes de voir le processus du raisonnement de l'homme de loi, et le système juridique lui-même, déformé par les contraintes inhérentes à l'informatique, et il plaidait pour que soient poursuivies des recherches sur ces processus, affirmant au surplus que les systèmes d'intelligence artificielle analogues à TAXMAN dont nous parlerons plus loin, ouvraient « des avenues plus prometteuses que les systèmes actuels », plus orientés sur la documentation, que vers l'information⁸⁴.

82. Cf: C. WOODY *et als*, « A Subject — Content — ... », *supra*, note 35, p. 31-32.

83. P. SLAYTON, *La recherche documentaire électronique dans les sciences juridiques*, Ottawa, Information Canada, 1974; aussi: « Electronic Legal Retrieval — The Impact of Computers on a Profession », (1973) 14 *Jurimetrics Journal*, 29-40.

84. *Id.*, *La recherche documentaire...*, *supra*, note 83, p. 29; « Electronic Legal Retrieval... », *id.*, p. 39.

David Henkel, ayant constaté aussi que le formalisme de la requête booléenne peut déformer le raisonnement du juriste, réclame également que des recherches plus approfondies soient réalisées sur ce sujet complexe⁸⁵. Y a-t-il enfin un jurimètre sérieux qui ne réalise les limites des systèmes de repérage actuels au plan de la recherche conceptuelle, et qui ne souhaite que l'on conçoive le plus tôt possible des systèmes qui complètent les modèles présents, en ajoutant à leur fonctionnement le véritable raisonnement par le processus analogique au simple repérage documentaire par la reconnaissance des formes linguistiques ?

Au cours des prochains paragraphes, nous allons cheminer vers la découverte des systèmes intelligents qui réaliseraient notre désir de voir le raisonnement juridique automatisé à un point qui aiderait le juriste à porter ses perceptions plus loin encore, en compagnie de ceux qui ont déjà tenté l'aventure vers l'inconnu en bâtissant de tels systèmes, ou en suggérant des moyens de discipliner le processus du raisonnement juridique. Nous terminerons finalement notre périple dans les univers habités par JUDITH, TAXMAN ou SHRDLU, en tentant de définir les paramètres d'un monde analogue qui soit nôtre, plus français que saxon peut-être, mais néanmoins aussi ordonné, cybernétique et performant qu'on puisse le désirer.

2.1. L'apprentissage par la logique.

La structuration du message que saisit celui qui s'informe, doit passer nécessairement pour sa rétention finale par diverses étapes de formalisation, au plan syntaxique entre autres. C'est à ce problème que s'est attaqué un professeur de l'Université Yale, à qui l'on doit une théorie dite de « normalisation » du langage juridique, qu'il a maintes fois décrite et expliquée dans divers congrès⁸⁶ et nombreux textes publiés aux États-Unis⁸⁷.

85. W. D. HENKEL, « Computer in Legal Research : More than Pushing Buttons », (1973) 14 *Jurimetrics Journal*, 10-18.

86. Par exemple, à la deuxième conférence nationale sur la recherche juridique automatisée, tenue par le Comité de droit et technologie de l'Association du Barreau américain ; voir : D. T. LINK and L. E. ALLEN, « Legislative Information System », in *Sense & System in Automated Law Research*, *supra*, note 11, pp. 125-137.

87. *Inter alia* : L.E. ALLEN, « Logic, Law and Dreams », (1959) 52 *Law Library Journal*, 131-144 ; « Symbolic Logic : a Razor-Edged Tool for Drafting and Interpreting Legal Documents », (1957) 66 *Yale Law Journal*, 833 ; « Beyond Document Retrieval Toward Information Retrieval », (1963) 47 *Minnesota Law Review*, 713-767 ; avec M.E. CALDWELL, « Modern Logic and Judicial Decision Making », (1963) 28 *Law & Contemporary Problems*, 213 ; « Logic — Language — Law », in *Computers and the Law*, R.P. Bigelow (ed.), Chicago, C.C.H. 1966, pp. 79-84 ; « Analysis of Law by Symbolic Logic », in *Computers and the Law*, R. P. Bigelow (ed.), 2nd ed., Chicago, C.C.H., 1969, pp. 167-174 ; « Normalization Approach to Information Retrieval in Law », in *Libraries and Librarianship*, 1967.

La théorie d'Allen se trouve la *ratio* du système ABF de l'*American Bar Foundation*⁸⁸, et surtout, à la source de fort nombreux commentaires, si bien qu'il importe qu'on s'y arrête pour une brève étude, même si d'autres types de formalisation des règles juridiques, tels les graphes, les logiques déontique⁸⁹ ou hohfeldienne⁹⁰, l'application des tables de décision ou la systématisation par ordinogrammes devraient également attirer notre attention⁹¹.

2.1.1. La normalisation des messages

La contribution de Layman Allen à la structuration du message légal restera sans doute longtemps importante et remarquable parce que ce théoricien se distingue en ayant été l'un des premiers juristes à s'attaquer de front au problème absolument crucial du langage juridique⁹². Au stade primaire de sa conception, avant d'être appliquée à des situations concrètes, la règle de droit est langage⁹³. Elle deviendra ensuite processus⁹⁴, mais elle est

88. Voir *infra*.

89. La logique déontique prend le droit substantif comme son sujet d'étude, qualifiant les règles du droit comme pertinentes aux foncteurs d'*obligation*, d'*autorisation* ou d'*interdiction*. Nous renvoyons ici notre lecteur aux travaux de messieurs Kalinowski, Perelman et Mehl.

90. Hohfeld prétendait pouvoir réduire le fonctionnement de la machine juridique à huit communs dénominateurs, les notions fondamentales du Droit: W.N. HOHFELD, «Some Fundamental Legal Conceptions as Applied in Judicial Reasoning», (1913) 23 *Yale Law Journal*, 16-59. Les développements de l'application de l'informatique et de la logique moderne au Droit ont ravivé l'intérêt des juristes pour les théories d'Hohfeld; voir: A.R. ANDERSON, «The Logic of Hohfeldian Propositions», (1971) 33 *University of Pittsburgh Law Review* 29; L. E. ALLEN, «Formalizing Hohfeldian Analysis», (1974) 28 *Southern California Law Review*, 428.

91. Nous tirons cette série d'exemples de H. BAUER-BERNET, «Effect of Information Science and the Formation and Drafting of Law», (1974) 14 *Jurimetrics Journal*, 235-239.

92. William Boyd, constatant la stagnation des méthodologies de la recherche documentaire automatisée, attribue cet immobilisme pour une bonne part au fait que les concepteurs de ces systèmes essayent de contourner plutôt que de vraiment régler le problème du langage juridique: W. E. BOYD, «Law in Computers and Computers in Law: a Lawyer's View of the State of the Art», (1972) 14 *Arizona Law Review*, 267-311 (277).

93. Nous gardons ici au terme *langage* sa signification la plus simple de «support d'information formé d'éléments de structure déterminée»; voir: L. COUFFIGNAL, *La cybernétique*, Paris, P.U.F., 1968, p. 37.

94. Appliquée à une situation concrète, la règle de droit atteint son objectif en devenant agent de création du comportement recherché par son auteur.

d'abord message⁹⁵, composée de mots⁹⁶, structurée par sa syntaxe⁹⁷, porteuse ultime de sémantique⁹⁸.

Comme on l'a vu déjà, nos systèmes de repérage automatisé actuels maîtrisent déjà les mots/signes⁹⁹, et nous verrons plus loin qu'il est raisonnable d'espérer un certain contrôle des sémantiques. Pour l'instant, nous restons avec Allen au stade de l'analyse syntaxique, une étape tout de même importante puisqu'elle nous permettra plus loin d'« écouter »¹⁰⁰ les mots perçus par nos systèmes de repérage intelligents.

Portée à ce point de la conception des systèmes de repérage des années télématiques, l'analyse d'Allen apporte une dimension utile à la maîtrise des règles/messages du droit. Dès la fin des années '50¹⁰¹, ce théoricien fécond avait remarqué déjà que les systèmes de repérage automatisé que nous connaissons encore aujourd'hui, pouvaient conduire à une impasse logique puisqu'ils ne permettaient pas d'analyse du texte produit, d'autant plus que celui-ci est apporté à l'ordinateur dans sa forme la plus brute, que le système informatique traditionnel ne peut logiquement décoder.

C'est pour surmonter cette difficulté qu'Allen conçoit d'« aider » l'ordinateur en annotant, ou normalisant les textes, de telle sorte que le système puisse par après suivre la piste qui lui est ainsi tracée par les opérateurs logiques¹⁰² dont le médiatiseur a balisé au besoin le message original¹⁰³.

95. La règle de droit constitue un « groupe fini, ordonné, d'éléments de perceptions puisés dans un "répertoire" et assemblés en une structure », (voir : A. MOLES, *Art et ordinateur*, Paris, Casterman, 1971 ; au même effet : J. CLOUTIER, *L'ère d'Emerac*, *supra*, note 62, p. 230).

96. Les mots constituent pour nous les signes élémentaires de la règle/message du droit.

97. La syntaxe établit les rapports des signes entre eux (cf : G. MOUNIN, *Clefs pour la sémantique*, Paris, Seghers, 1972, p. 40).

98. Le terme *signification* serait-il ici plus approprié ? Nous voulons exprimer par *sémantique*, la signification ou l'information qu'aura retenue le récepteur des messages après avoir décodé les éléments informatifs perçus (cf : A. MOLES, *supra*, note 95, p. 156).

99. On se souviendra que leur repérage est fondé sur la reconnaissance de mots clés déterminés.

100. Ce terme garde le sens que nous avons décrit déjà.

101. Voir par exemple, « Beyond Document Retrieval... », *supra*, note 87.

102. Allen retient pour échafauder sa théorie cinq opérations logiques, traduites par des opérateurs et des notations empruntées aux techniques de représentation de la logique formelle. Ces opérations sont :

l'inférence (IF... THEN),

la « disjonction » (OR),

l'inclusion (AND),

l'équivalence (IF and ONLY IF),

la négation (NOT).

Plus simplement, l'opérateur OU représente l'*union* et l'opérateur ET l'*inclusion* de la logique booléenne.

103. En plus des renseignements qu'on pourra extraire des textes publiés par L. ALLEN lui-même (voir, *supra*, note 87), on pourra voir également les commentaires et explications

Ayant donc préalablement réduit ce message en un assemblage de propositions constitutives du tout, et l'ayant normalisé par l'addition des opérateurs logiques nécessaires placés entre les propositions elles-mêmes représentées maintenant par des symboles, le logicien, une fois son raisonnement formellement complet, remplace cette fois les symboles par les propositions, seuls les opérateurs logiques restant apparents. Le texte original devient donc ainsi formellement normalisé, dans la matrice logique des opérateurs prédéterminés.

La rigueur sans complaisance de cette méthodologie constitue l'atout substantiel majeur de ce jeu logique. La mécanique des inférences qu'elle génère, s'applique avec un automatisme presque machiavélique. Fondamentalement booléen, ce type de raisonnement exclut le rôle de l'intention, par ailleurs toujours si chère au juriste traditionnel, qui n'a d'ailleurs pas jusqu'ici ménagé ses critiques à l'endroit de la théorie d'Allen.

On reproche ainsi à la normalisation de ce théoricien d'obliger le juriste qui désire en tirer parti, à apprendre un langage fort compliqué, ce que ce dernier n'est pas nécessairement enclin à faire¹⁰⁴, quand ce ne serait qu'en raison de ses autres occupations souvent plus pressantes à court terme. Cette objection, si elle est uniquement formelle, ne manque pas d'intérêt ! Elle est même à peu près convaincante à elle seule pour expliquer la réticence de nos législateurs à en faire usage. On le voit mal en effet rédiger une loi d'exception urgente en en réduisant d'abord les termes et éléments en propositions logiques annotées d'opérateurs soigneusement choisis. Le citoyen risquerait d'ailleurs de se perdre sans espoir dans un texte produit de façon aussi ampoulée.

Il n'est pas sûr d'ailleurs que la normalisation d'Allen règle vraiment les problèmes les plus importants que génère la règle de droit. Daniel Jacoby fait ainsi remarquer avec justesse que les ambiguïtés syntaxiques que prétend effacer la normalisation d'Allen, ne génèrent en fait qu'un nombre de problèmes restreint, ne suscitant que peu de contentieux¹⁰⁵. L'observation se révèle fort intéressante, rappelant encore une fois que le vrai problème fondamental du droit, celui de la qualification, se situe bien au niveau de la sémantique, et bien peu ailleurs, si ce n'est au plan de la perception des règles de droit. Or, là encore la normalisation d'Allen prête flanc à la critique, si on en exagère la portée véritable.

fournis, *inter alia*, par W. E. BOYD, *supra*, note 92, et par T. H. EDWARDS et J. P. BARBER, « A Computer Method for Legal Drafting Using Propositional Logic », (1975) 53 *Texas Law Review*, 965-1003.

104. Voir: BOYD, *supra*, note 92, p. 278.

105. D. JACOBY, « La composition des lois », (1980) 40 *Revue du Barreau*, 3-41, à la page 33.

Il reste en effet qu'une bonne partie des règles de droit, si souvent non écrites, parties de la culture générale des peuples, souffrent difficilement qu'on les réduise en propositions logiques¹⁰⁶. Le vécu du droit n'est pas codifié, et pourtant il conditionne l'administration de larges portions de nos règles juridiques au plan de la définition des droits civils par exemple, où l'interprétation des tribunaux a constitué, à partir du vécu des sociétés, le vrai code de nos libertés.

La logique, froide et impersonnelle, pourrait ordonner le contenu, même factuel de notre héritage jurisprudentiel, mais au prix d'une désincarnation des normes qui risquerait bien de créer un droit purement mécanique dont nos droits fondamentaux s'accommoderaient fort mal. Le logicien lui-même est plus dangereux que ses systèmes et ses machines, bien que son esprit de systématisation puisse nous apporter d'indéniables bénéfices. En dépit des critiques que nous venons de soulever à propos de la normalisation d'Allen, ce dernier confirme en effet la règle générale et nous laisse des éléments forts positifs de sa théorie.

C'est ainsi que l'analyse syntaxique dont elle tend à faciliter la réalisation, constitue l'une des étapes essentielles à la constitution des systèmes intelligents auxquels nous aspirons. Nous en avons besoin en effet pour décoder les interventions du requérant, dont les questions et réponses doivent être au moins « écoutées » par la *machina sapiens*. La normalisation d'Allen peut aider au minimum à localiser et identifier les propositions que le système global devra traiter pour construire ses raisonnements. À ce point précis, les opérateurs logiques utilisés par Allen pourraient s'avérer fort efficaces.

Ces opérateurs stimulent d'ailleurs notre imagination sous divers aspects. Ils ne sont pas aussi sans nous rappeler les instructions¹⁰⁷ des langages informatiques¹⁰⁸, charnières des énoncés¹⁰⁹ des programmes dont

106. Voir encore BOYD, *supra*, note 104.

107. Une instruction est « une unité d'information particulière, interprétée comme une opération à effectuer ». On notera que cette instruction comporte deux parties, un *code opération* (qui indique *quoi faire*) et un *code opérande* (qui désigne *où faire* l'opération commandée) (*Principes des ordinateurs IBM*, Paris, IBM France, p. 86). La première partie constitue l'essentiel de l'*instruction* si bien que nous serons porté plus loin à employer l'un pour l'autre les termes qui les désignent.

108. Les « grands » langages informatiques sont COBOL, FORTRAN, PL/I, APL, BASIC, etc. (On définit le langage symbolique comme « tout ensemble de symboles, avec ses règles d'emploi, utilisé dans un programme pour représenter des codes opérations, des fonctions et des adresses ». (Voir : *Principes des ordinateurs IBM*, *op. cit.*, *supra*, note 107, p. 115).

109. L'énoncé d'un programme est constitué de l'ensemble appliqué à une situation donnée, d'une instruction et de son adresse individualisée. Une série d'instructions en BASIC se lit par exemple ainsi : 100 IF P(N) = 0 THEN GOTO 200. Yves Leclerc explique cet énoncé (voir : *L'ordinateur individuel*, *op. cit.*, *supra*, à la p. 209). Déjà on aura remarqué les instructions GOTO et IF... THEN.

originent¹¹⁰, la substance même de ces langages et l'objet de leur compilateur¹¹¹.

De cette observation jusqu'à l'inférence subséquente qui entraîne à formuler l'hypothèse qu'il pourrait être possible de concevoir un langage propre au juriste, construit sur des instructions (ou opérations) qui seraient pour une bonne part celles qu'utilise Allen, puisque tous les langages de programmation assignent des significations précises et non ambiguës à des mots comme SI (IF), ET (AND), OU (OR) ou AINSI (THEN)¹¹², il n'y a plus qu'un pas à franchir ! Or, la théorie d'Allen prouve ici sa fécondité, car l'*American Bar Association Foundation* a franchi ce pas en élaborant son processeur ABF, le premier des quatre systèmes que nous allons maintenant étudier brièvement à tour de rôle.

2.1.2. La structuration des processus

De ces quatre systèmes, les trois derniers prétendront à l'intelligence, alors que le premier ne se réclamera sans doute que de l'ingéniosité. Malgré sa modestie relative, ce dernier logiciel nous enseignera néanmoins les pratiques vertus de la structuration des formes et même de la normalisation des textes chère à Layman Allen. Peut-être pourra-t-il même suggérer à nos législateurs des applications qui faciliteraient éventuellement certaines de ses tâches de rédaction.

Avant TAXMAN, précédant MAC et même JUDITH, commençons donc notre revue de systèmes par le processeur ABF d'*American Bar Foundation*.

2.1.2.1. Le processeur ABF

C'est à l'action conjuguée du juriste James Sprowl, de l'expert linguiste Barbara A. Sangster et de l'informaticien Gregory A. Suski, soutenue par l'apport financier provenant de l'*American Bar Foundation*, que nous devons

110. Le programme origine est « le programme codé dans un langage autre que le langage-machine, et qui doit être traduit en langage-machine avant d'être utilisé ». (*Principes...*, *supra*, note 107, p. 115).

111. Le compilateur est « le programme de traduction qui traduit les macro-instructions d'un programme symbolique en instructions symboliques « une pour une » et qui transmet cette séquence d'instructions à un programme d'assemblage en vue de leur traduction en langage-machine ». (*Ibid.*).

112. J. SPROWL, « Automating the Legal Reasoning Process... », *infra*, note 113, p. 11.

cet ensemble ABF¹¹³, dont l'architecture globale s'échafaude sur les deux pièces maîtresses du *processeur*¹¹⁴ qui traite l'information, et du *langage* normalisé qui structure formellement le dernier objet.

— *Description du système*

Au départ, les auteurs d'ABF avaient conçu l'idée qu'il serait possible, grâce à un processeur spécialement articulé et à condition de normaliser préalablement la donnée-objet, de construire un système informatisé qui produirait automatiquement des documents légaux à partir d'un texte pré-enregistré en mémoire de l'ordinateur. Dûment préparés, ces documents revêtiraient ainsi une forme telle, que le système serait entraîné à guider le processus de raisonnement du rédacteur/utilisateur, en lui posant même au bon moment des questions pertinentes, propres à l'amener à poser des choix corrects lorsqu'il individualise pour ses fins propres les modèles documentaires soumis. Le système ABF fonctionne ainsi de la manière suivante :

1. À partir de documents pré-enregistrés, qui sont en l'espèce des formules contractuelles (v. g. un testament, un acte de fiducie ou un bail complexe) assorties de « blancs » permettant l'insertion des noms de parties, des montants dus, etc.¹¹⁵, l'utilisateur d'ABF, un rédacteur de documents, se voit entraîné à remplir la formule proposée par le système, en l'adaptant à ses besoins avec l'aide du processeur, qui lui amène, en lisant les formules préparées, des questions provoquées par des commandes spéciales, parties du logiciel opérant, portant par exemple sur le nom des parties, le lieu de rédaction des contrats, le choix de certaines clauses, etc.
2. L'action globale du processeur complète sa fonction simple de processeur d'informations, par une action d'*éditeur*, qui l'entraîne ensuite à structurer la forme des documents rédigés et à les produire à son auteur.

113. J. A. SPRAWL, « Automating the Legal Reasoning Process: A Computer That Uses Regulations and Statutes to Draft Legal Documents », (1979) *American Bar Foundation Journal*, 1-81. Dans les paragraphes qui suivent, nous allons tirer nos renseignements de ce texte généralement, sans le citer de nouveau, sauf plus loin, à quelques reprises.

114. J. SPRAWL ne définit pas ce qu'il entend exactement par le terme *processor*. ABF, comme processeur, nous paraît être un programme de traitement et d'ordonnancement de l'information, qui remplit en même temps des fonctions de compilateur.

115. Notons ici en passant qu'ABF offre à ses utilisateurs la possibilité d'effectuer certaines opérations de calcul, permettant par exemple de compter les intérêts composés ou les coûts de crédit découlant d'une transaction de vente où le prix n'aurait pas été entièrement versé au moment de la conclusion du contrat.

Comme on le voit, ABF constitue au point minimal, un ingénieur automate¹¹⁶ éditeur. L'intelligence apparente du robot¹¹⁷ ne traduit cependant que l'ingéniosité créatrice de son auteur, sans démontrer par conséquent des processus raisonnables qui lui appartiennent en propre.

ABF semble toutefois posséder comme nous l'avons dit plus haut, un langage qui lui soit personnel. Mieux encore, il pose des questions ! Serait-il donc intelligent ? Pour résoudre ce problème, il nous faut ici ouvrir sa mémoire et observer le processus par lequel il s'exprime. S'il est autonome¹¹⁸, il nous faudrait conclure à l'intelligence d'ABF.

— *Le langage d'ABF*

Suivant ce que nous avons dit déjà, la mémoire d'ABF retient des documents « contractuels » qu'on a d'ailleurs eu l'heureuse idée de construire en fonction des textes légaux qui en constituent le fondement juridique. Ces unités documentaires/sources, que l'utilisateur doit individualiser pour répondre à ses besoins propres, ont été non seulement assorties des « blancs » nécessaires à cette fin, mais également logiquement structurées, ou *normalisées*, en propositions. Rattachées à la structure d'ensemble du document

-
116. R. MALONE définit l'*automate* comme « [A] machine often in human form, incorporating mechanisms that perform without human intervention » (in, *The Robot Book*, New York, Push Pin Press, 1978, p. 158). En rédigeant cette définition, Malone a surtout à l'esprit les automates de Vaucanson, qui sont des mécanismes physiques reproduisant les gestes d'un être vivant (v. g. le coq de l'horloge astronomique de la cathédrale de Strasbourg ou le canard de Vaucanson). Nous nous plaçons toutefois ici dans la position du cybernéticien pour qui une *machine* peut se manifester comme un système par ailleurs abstrait, mais susceptible de produire néanmoins des résultats. Avec SKYVINGTON, nous dirons ainsi « que l'automate est un dispositif formel abstrait qui, en fonction de l'*input* qu'il reçoit, et de l'état dans lequel il se trouve, (a) produit un *output* et (b) change d'état : *supra*, note 39, p. 187.
117. Robert Malone définit le robot comme « a machine that simulates the function or appearance of a human being » : *supra*, note 116, p. 158. Encore une fois, le robot cybernétique n'a pas besoin d'être actualisé physiquement pour exister de plein droit. Shrdlu, avec lequel nous lierons plus loin connaissance, n'est qu'une image qu'on voit agir sur l'écran cathodique d'un terminal. Lorsqu'il simulera l'homme, notre robot cybernétique le fera en imitant le processus raisonnable qui caractérise l'*homo sapiens*.
118. La notion d'*autonomie* est absolument capitale dans le débat sur l'intelligence artificielle. Ne peut être intelligente que l'action entreprise sous l'empire du libre arbitre de son auteur, et en fonction d'objectifs qu'il a lui-même fixés. Le scribe automate de Pierre et Henri-Louis Jacquet-Droz (voir : *The Robot Book*, *supra*, note 117, p. 35) n'écrit pas parce qu'il en a ainsi décidé, mais parce qu'un mécanisme dont il est totalement dépendant et qu'on a mis en marche sans aucune intervention de sa part, lui fait plier les doigts et bouger la main en faisant tous les gestes de l'écrivain. Nos logiciels de repérage documentaires actuels, tel STAIRS par exemple, sont des robots, parce qu'ils reconnaissent des formes linguistiques numérisées non de leur propre chef, mais parce que le mécanisme de leur logique le veut ainsi.

par le déroulement global du processus, et articulées sur le fonctionnement d'opérateurs de type booléen, elles forment en définitive le modèle normalisé de Layman Allen¹¹⁹. En opérant le système ABF, son utilisateur se trouve ainsi conduit à procéder suivant une démarche qui autovérifie continuellement l'exactitude de son déroulement, puisque le système produit automatiquement à chaque fois qu'une décision doit être prise, autrement dit lorsqu'un « blanc » doit être rempli, une question pertinente à la substance du problème traité, grâce à l'action de l'opérateur logique que repère le processeur à la lecture séquentielle du document traité. Avant de produire par exemple, la déclaration d'une action en revendication d'un bien vendu impayé, le système demanderait ainsi à l'utilisateur du système si les biens dont il s'agit ont été livrés.

Parce que le processeur ABF peut générer automatiquement des questions pertinentes conduisant son utilisateur à poursuivre une démarche logiquement orchestrée, les auteurs d'ABF proclament que ce processeur génère ses propres programmes, ce qui ressemble à première vue à un processus d'intelligence artificielle¹²⁰. Une simple observation de ce type de traitement nous entraîne cependant vers des conclusions contraires à cette prétention.

Comme Eliza, ABF « oit » et « écoute », mais sans « entendre » ni « comprendre ». Seule la reconnaissance formelle des opérateurs prédéfinis du système, lui fait générer automatiquement, presque mécaniquement, les questions qu'il produit ensuite au rédacteur.

L'opération reste donc toujours absolument formelle. ABF ne jouit pas de faculté cognitive. Comme Eliza, il perçoit sans connaître. Toute appréhension conceptuelle lui échappe et, *a fortiori*, toute possibilité de raisonnement quelconque. ABF reste limité à la structuration automatique des formes et à la simulation du processus dynamique du raisonnement intelligent auquel il ne peut prétendre, puisqu'il ne peut ni analyser, ni abstraire, ni généraliser.

Malgré tout, il faudrait bien se garder de faire la fine bouche à l'endroit d'ABF ! Ce logiciel remplit sans doute son rôle d'automate avec efficacité et compétence, produisant les documents attendus avec promptitude et suivant les normes établies.

119. Le langage ABF (le volet logique du système global) opère de plusieurs façons, mais fondamentalement à partir de la série logique IF... THEN... [OTHERWISE]. On y retrouve ainsi en bonne place les opérateurs AND, OR, NOT, et diverses variantes de ces instructions de base, tels les IFF, ENDIF ou EXOR.

120. On pourrait croire en effet ABF autonome dans le sens où nous l'entendions, *supra*, en note 118.

ABF traduit en fait dans la réalité une conception documentaire générale fort ingénieuse, qui s'appuie sur une saine garantie de fiabilité et de sécurité juridique puisque les textes qu'il traite proviennent autant que possible de la phraséologie même de la loi. Sa structure logique reste aussi en principe impeccable et fondée sur une méthode dont on peut mettre en doute la rentabilité mais non la qualité. Il me semble enfin que l'hypothèse qu'on a retenue, de construire un langage informatique à partir des opérateurs logiques d'Allen, pourrait s'avérer féconde si on poussait les prémisses de ce raisonnement à leur ultime.

En concevant l'architecture générale du projet MØREL ¹²¹, il nous était apparu au départ que les termes juridiques fondamentaux qui constituaient nos modèles légaux pourraient peut-être se transformer de quelque manière en *instructions* ¹²² informatiques, armatures d'un langage processeur nettement juridique, un peu comme l'est aujourd'hui ABF. MØREL aurait pu devenir de la même manière un langage rédacteur, un automate légiste. Le terme *QUORUM*, transformé en *instruction* logicielle, ferait générer un ou plusieurs textes sur les normes à imposer quant au quorum des assemblées d'actionnaires, de membres de comité, d'administrateurs de sociétés publiques, para-publiques et de toutes natures.

N'agissant encore qu'en raison d'un mécanisme de reconnaissance des formes, MØREL aurait donc été lui aussi plus objectivement ingénieux que subjectivement intelligent. Comme ABF, il n'aurait pas été capable non plus d'inférences autonomes, générées par les rouages d'une logique interne activement indépendante de ses mécanismes opérationnels.

Les systèmes dont nous étudierons maintenant le fonctionnement vont prétendre aux capacités de généralisation du concept et d'analogie dont n'était pas nanti l'ensemble ABF. Avant d'aborder MAC et TAXMAN, commençons donc par JUDITH, que nous aimerons peut-être autant pour son intelligence que pour ses formes.

2.1.2.2. JUDITH

Partant comme Buchanan et Headrick ¹²³ du principe que l'informatique juridique n'a de véritable avenir que si elle s'insère dans un

121. On pourra voir par ailleurs les objectifs plus rapprochés que nous nous étions fixés dans l'ouvrage de notre collègue P. ISSALYS, *Langage et système des lois*, Québec, Conseil de la langue française, 1980, p. 10. Nous devons à Pierre Issalys la réalisation finale du projet MØREL auquel il a donné avec raison une orientation différente, et qui se présente aujourd'hui comme cette riche étude prospective sur l'organisation administrative que nous venons de citer.

122. L'*instruction* à laquelle nous référons ici, s'identifie naturellement à celle que nous avons définie déjà, *supra*, note 107.

123. B. G. BUCHANAN and T. E. HEADRICK, « Some Speculations about Intelligence and Legal Reasoning; (1970) 23 *Stanford Law Review*, 40-62.

ensemble plus considérable et si elle s'oriente vers le contrôle du raisonnement juridique au lieu de se cantonner au simple rôle d'outil de repérage documentaire, deux chercheurs allemands, Walter G. Popp et Bernhard Schlink¹²⁴, juristes rattachés respectivement aux universités d'Heidelberg et de Darmstadt, ont réalisé avec l'aide de professeurs de l'université Stanford un logiciel appelé JUDITH¹²⁵, capable, semble-t-il, de produire par les seuls moyens de ses connaissances acquises et de sa logique interne, des traitements de concepts équivalents à des raisonnements juridiques.

Notre rencontre avec JUDITH apparaissant prometteuse, penchons-nous donc sur la description de ses formes et arrêtons-nous au cheminement de sa logique pour voir si l'ensemble qu'on nous propose est vraiment intelligent, et susceptible d'être marié un jour à une institution de même nature et qui nous serait vraiment propre.

— Description de JUDITH

On peut lier connaissance avec JUDITH en engageant la conversation avec cet accueillant système, par la voie d'un terminal qui fera parvenir des réponses aussi pertinentes que possible en un mode au moins interactif, au juriste qui s'est amené jusqu'à son clavier d'accès.

Il semble bien que tous ceux qui ont recours à JUDITH, le font d'ailleurs parce qu'ils ont des problèmes. Suivant son option d'entrée principale (le *case option*)¹²⁶, c'est ainsi assez étrangement par la soumission à JUDITH du remède proposé par le juriste en difficulté, que cette sage intelligence aidera ensuite ce dernier à construire un sain raisonnement l'amenant à la résolution de son problème. Comme le système est pour le moment spécialisé en responsabilité civile, l'utilisateur, après avoir d'abord lui-même qualifié son problème comme étant par exemple une *negligence*¹²⁷ et jugé qu'une réclamation en dommages lui apparaissant le remède le plus efficace à ses difficultés, soumettra ainsi ce terme *damages* à JUDITH, qui, dès lors, lui indiquera des *causes d'actions* susceptibles de générer des *actions en dommages* comme celle proposée par le requérant. Au fur et à mesure que

124. Nous verrons plus loin que Meldman et McCarthy manifesteront des préoccupations identiques, en nous expliquant le fonctionnement de leurs systèmes MAC et TAXMAN.

125. W. G. POPP and B. SCHLINK, « JUDITH, A Computer Program to Advise Lawyers in Reasoning a Case », (1975) 15 *Jurimetrics Journal*, 303-314. Nous tirons nos renseignements sur JUDITH principalement de ce texte, auquel nous référons maintenant de façon générale.

126. L'autre mode d'accès que permet JUDITH est le *specify option*, qui permet de s'intégrer au système à un point donné de sa structure, sans passer nécessairement par toutes les étapes dont son déroulement est constitué.

127. Pour ne pas fausser l'explication de JUDITH, nous ne changerons pas ici la référence au *tort* spécifique du *Common Law* auquel le texte de Popp et Schlink fait référence.

des concepts de droit reliés lui seront présentés, ce dernier pourra interroger le système sur les éléments essentiels de ces notions, et il pourra vérifier de cette façon si la qualification première qu'il avait faite de son problème était correcte, et, de là, si son choix de remède s'avérait le meilleur possible.

Il va sans dire qu'une telle approche risque de désarçonner au premier abord un juriste cartésien qui préférerait sans doute procéder à partir des principes de droit jusqu'à la résolution des problèmes, plutôt que d'adopter cette démarche inversée qui le ramène de la résolution du problème spécifique jusqu'aux normes légales. Ce type de démarche appartient plus au processus de raisonnement du juriste de *common law* qu'à celui de l'avocat formé dans un environnement de droit codifié, qu'il soit français ou allemand. La construction de cette approche surprenante s'explique sans doute du fait que le projet a été réalisé à Stanford entre 1970 et 1975, et qu'une partie du logiciel et de la conception fondamentale du système a été inspirée peut-être par le logiciel *Heuristic DENDRAL* dont traitent ailleurs Buchanan et Headrick¹²⁸. Notre perplexité reste cependant presque entière malgré ces explications plausibles, parce que l'architecture globale de JUDITH présente autrement des petits airs de famille cartésiens qui ne vont pas déplaire aux juristes francophones que nous sommes, et que nous ne retrouvons pas chez MAC, ni chez TAXMAN.

— La structure de JUDITH

Suivant ce que nous rapportent Popp et Schlink à ce sujet, il semble bien que le fonctionnement de JUDITH provienne de l'action concertée de deux fichiers interreliés, le premier contenant une liste des prémisses individuelles classifiées par simple ordre d'entrée (*random order*), le second entreposant les structures des jeux de prémisses (*construction plans of sets of premises*).

À ce point de notre étude, ces prémisses et jeux de prémisses ne peuvent qu'attirer notre attention, car elles constituent l'un des éléments essentiels à un système intelligent. Il ne suffit pas en effet pour un tel système de manifester des capacités logiques. Il lui faut également être « instruit », c'est-à-dire pouvoir compter, face à un problème donné, sur un acquis préalable, un modèle prédéterminé, à partir duquel il puisse comparer la situation nouvelle qu'on lui soumet, pour pouvoir projeter ensuite les raisonnements et élaborer les conclusions qu'on en attend, ou qui lui permettra d'apprendre par la réception de nouvelles données, d'où il puisse faire émaner à nouveau d'autres constructions logiques lorsqu'il sera confronté à des problèmes

128. *Heuristic DENDRAL* est un logiciel permettant l'édification de raisonnement inductif à partir d'une base de données portant sur la chimie moléculaire; voir: BUCHANAN et HEADRICK, *supra*, note 123.

spécifiques inédits. Un système intelligent opère en d'autres termes aussi bien en fonction de ses connaissances préacquises qu'en raison de ses capacités logiques. Un tel système ne peut qu'enseigner ; il doit pouvoir également apprendre, si l'on veut qu'il puisse en retour raisonner.

Nous savons déjà que JUDITH s'exprime, mais nous devons nous assurer que cette verbalisation n'est pas celle d'un pur automate, mais d'un système qui puisse au moins réussir une analyse conceptuelle à partir d'une situation qu'on aurait soumise à son attention. Mémoire et calculateur sont donc à la fois nécessaires à JUDITH pour qu'on puisse conclure à son intelligence.

Après observation, il semble qu'on puisse se déclarer satisfait de JUDITH sur le premier plan au moins, puisque le système que nous étudions possède en principe comme préacquis une structure hiérarchique de type arborescent (semble-t-il) échafaudant l'édifice de la responsabilité civile. Chaque partie qui s'ajoute à cette construction, représente une notion qu'on y trouve définie par ses éléments constitutifs.

Une telle base de départ ne peut que plaire aux cartésiens que nous sommes, et nous réjouir d'autant plus que cet édifice n'est pas si rigide qu'il empêche le système d'élaborer des cheminements à la carte, adaptés aux problèmes spécifiques qu'on lui soumet, car n'oublions pas que le système doit raisonner en mode autonome, si l'on veut pouvoir conclure à son caractère intelligent. Ayant constaté la mémoire et les facultés logiques de JUDITH, soumettons notre système à ses derniers tests pour nous convaincre, le cas échéant, de ses capacités intellectuelles.

D'après les théoriciens de l'intelligence artificielle, un système doit être capable, comme l'*Homo sapiens*, de résoudre un problème en 1) le réduisant en plusieurs petits sous-problèmes, et 2) en étant capable de résoudre chacun de ceux-ci à partir de règles décisionnelles claires et cohérentes¹²⁹.

Nous pouvons constater facilement de la structure des prémisses que nous connaissons déjà, que JUDITH n'éprouve pas de difficultés à morceler un problème complexe en plus petits ensembles interreliés, à propos desquels des décisions individuelles doivent être prises, ce qui remplit la première condition. Pour la seconde, Popp et Schlink nous affirment que c'est le compilateur de JUDITH, et lui seul¹³⁰, qui construit le cheminement de la démarche logique à la fois entreprise et conditionnée par le requérant, au fur et à mesure des réponses que ce dernier introduit dans le système et des décisions qu'il prend lui-même sur la pertinence des propositions que le système lui adresse.

129. Cf: BUCHANAN et HEADRICK, *supra*, note 123, à la p. 45.

130. POPP et SCHLINK, *supra*, note 125, à la page 310.

Si tel est le cas, JUDITH représente ainsi plus qu'un ingénieux agencement de formes. C'est aussi un cerveau qu'il faut admirer pour son intelligence, même si ses possibilités semblent bien limitées.

Nous croyons en effet que JUDITH ne pourra jamais apprendre grand-chose car ses capacités d'emmagasinage de données demeurent bien restreintes, sans compter que ses professeurs risquent bien de se lasser à l'effort considérable requis pour lui enseigner les principes fondamentaux du droit.

Coupé de ces schémas connus, notre système devient totalement incapable de réussir quoi que ce soit. Pis encore, même à l'intérieur de ses connaissances, JUDITH reste sans grande puissance créatrice, et les bornes de sa compétence sont rapidement atteintes une fois qu'on a épuisé les quelques instructions de son langage opérationnel. Après avoir réussi une analyse conceptuelle, JUDITH s'essouffle pour de bon. Elle en perd son FORTRAN ! C'est là une limitation sérieuse qui risque toutefois d'affecter aussi les compagnons de JUDITH.

MAC et TAXMAN souffrent-ils de semblables handicaps ? Commentons par visiter MAC.

2.1.2.3. MAC

Nous allons maintenant, avec MAC d'abord et TAXMAN ensuite, pouvoir cerner les limites et déterminer les vraies frontières de ce monde étrange de l'intelligence artificielle¹³¹, où la vie se déroule en exclusivité suivant des modèles préétablis composés d'agents rigoureusement prédéfinis, un micro-univers¹³² en dehors duquel rien n'existe, et à l'intérieur duquel les

131. Nous n'avons pas défini jusqu'ici le concept d'*intelligence* artificielle. Sans mener une étude exhaustive sur ce point, signalons que les auteurs s'entendent normalement pour y voir cette sous-discipline de l'informatique par laquelle on cherche à programmer un ordinateur de façon telle que le comportement subséquent de son logiciel traitera l'information suivant des processus apparemment raisonnables, analogues à ceux de l'*homo sapiens* (voir à cet effet, *inter alia*, M. A. ARBIB, *Computers and the Cybernetic Society*, New York, Academic Press, 1977, 281 ou 298 ; L. RASTRIGUINE et P. GRAVÉ, *La cybernétique telle qu'elle est*, Moscou, MIR, 1979, p. 189 et *sq.*, ainsi que SKYVINGTON, *supra*, note 39).

132. La notion de *micro-univers* est essentielle à la constitution de tout système d'intelligence artificielle, parce que *tous* les éléments du monde où évolue ce système, ainsi que leurs interrelations, doivent être précisément définis sans quoi les communications deviennent impossibles à l'intérieur du système. M. ARBIB, (*supra*, note 131, p. 303) définit le micro-univers comme « an environment in which a conversation can take place ». Pendant ce temps, à Berkeley, PAM (*Plan Applier Mechanism*) analyse de courts textes pour Richard Wilensky, alors qu'à l'université Yale, FRUMP (*Fast Reading Understanding and Memory Program*) rapporte les nouvelles du jour à son interlocuteur, Gerald F. Dejong (voir : F. HARROIS-MONIN, « "PAM" et "FRUMP" ou comment l'esprit vient aux ordinateurs », in *Science & Vie*, n° de mai 1980, p. 94).

acteurs ne suivent qu'une seule vocation. Les habitants de ces microcosmes ne traitent dès lors que d'un sujet. Certains de ces robots déplacent et empilent des blocs¹³³, alors que d'autres, plus intellectuels comme Abdul et BORIS¹³⁴, parviendront même à parler de politique internationale et à analyser de courts textes traitant de relations interpersonnelles. Nos personnages à nous se prétendront juristes et tenteront des analyses en fiscalité corporative dans le cas de TAXMAN ou en responsabilité civile pour MAC, dont nous analyserons les exploits en premier lieu.

Nous devons le système MAC à Jeffrey A. Meldman, un juriste diplômé du *Harvard Law School* et rattaché aujourd'hui au *Massachusetts Institute of Technology*¹³⁵. Réalisant à l'instar de Slayton, Popp et Schlink ou McCarthy que les systèmes classiques de repérage documentaire automatisé restaient incomplets parce qu'ils se bornaient à retrouver des unités documentaires purement formelles sans être capables de raisonner à partir de leur message substantiel, Meldman a formé à son tour le projet de mettre au point le prototype d'un système qui produirait une analyse juridique raisonnée découlant des résultats obtenus en insérant dans un système global de traitement de modèles un jeu de données factuelles, préalablement ainsi modélisé, comparé dès lors à un autre modèle, plus général, structurant des règles de droit, déjà accouplées ou non à des séries de fait préenregistrées par le système général.

Un tel projet se heurte dès le départ à deux ordres de problèmes également formidables, que l'auteur doit maîtriser de quelque façon. Il faut d'abord que le système puisse saisir les messages que son interlocuteur lui fera parvenir et, en retour, lui répondre de façon compréhensible, et il doit aussi être capable de structurer cette réponse dans un ordre substantiellement logique par rapport à ses connaissances anciennes ou nouvellement acquises par l'intervention même de l'interlocuteur. Notre système intelligent doit en d'autres termes pouvoir « écouter » et « comprendre »¹³⁶ pour arriver à raisonner et converser.

133. C'est là l'univers de SHRDLU, la créature de Winograd dont nous reparlerons plus loin.

134. Abdul est un système qui dialogue sur le sujet général de la Guerre des Six Jours, alors que BORIS, un système également, peut analyser des situations factuelles simples et même, semble-t-il, raisonner à partir des connaissances acquises des textes qu'il a lus, et qui lui ont rapportés les faits appris (voir: le numéro du 30 juin 1980 de *Newsweek*, p. 52).

135. J. A. MELDMAN, « A Structural Model For Computer-Aided Legal Analysis », (1977) 6 *Rutgers Journal of Computers and Law*, 27-71. Les renseignements que nous tenons du Projet dont on traitera maintenant, proviennent de ce texte auquel nous référons généralement.

136. Nous restons fidèle toujours à l'échelle Schaeffer rapportée précédemment; voir, *infra*, note 43.

Au cours des quelques paragraphes qui vont suivre, nous ne pouvons pas naturellement considérer la myriade de difficultés que soulèvent ces deux actions apparemment simples, ni tenter une explication des types de résolutions possibles qu'on peut apporter à ces questions, mais nous allons tenter néanmoins d'esquisser dans leurs grandes lignes les solutions retenues par Jeffrey Meldman, pour que son système puisse *écouter* et *comprendre* dans un processus voulu raisonnable par son créateur.

— *Le système écoute*

Sur ce plan, le système artificiel copie fidèlement l'*Homo sapiens* puisqu'il ne peut saisir qu'un langage dont il possède les clés de décodage. On n'arrivera donc à communiquer avec le monde artificiel que par un langage lui-même créé de toutes pièces en fonction de l'ordinateur et, parfois, en raison aussi de l'interlocuteur humain qui peut ainsi parler dans sa langue au système qui sera équipé d'un traducteur approprié.

Dans le cas qui nous intéresse présentement, un tel traducteur n'existe pas cependant et l'homme doit par conséquent s'adresser au système en langage PSL (*Preliminary Study Language*).

Ce compilateur est conçu à partir d'un sous-ensemble interne qui correspond à un modèle linguistique appelé *case grammar* qui joue ici pour nous un rôle capital. *Case grammar* ajoute en effet aux capacités d'analyse grammaticale de PSL, cette perspective fondamentale qui l'entraîne à considérer une phrase comme une *action* (manifestée ordinairement par le verbe), autour de laquelle gravitent un *agent* (représenté par le sujet) et un *objet* (qu'on retrouvera dans le complément direct de la phrase). En raison de *case grammar*, il faudra donc faire passer la forme de toute communication avec MAC, par la structure et sous l'aspect requis par le modèle linguistique, et dûment traduire au surplus ces propositions conformément aux notations du langage PSL¹³⁷.

137. Le lecteur trouvera au texte de Meldman, cité *supra*, note 135, les illustrations nécessaires à sa compréhension. Rapportons simplement ici cette notion essentielle à l'effet que l'ensemble PSL — *case grammar* s'exprime en termes de *relations*. L'une des plus importantes est *kind of*, absolument indispensable pour que le système puisse établir des analogies. Il lui est autrement impossible de conclure par lui-même qu'une arme à feu est un objet. Quelqu'un doit donc établir la relation *kind of* entre *arme à feu* et *objet*, pour que le système puisse faire l'analogie. Cette relation deviendra dès lors;

	<u>kind</u>	
arme à feu		objet
	ou	
<u>kind</u>	objet	arme à feu ;

signifiant au système qu'une arme à feu est un objet.

On aura compris, et nous n'insisterons pas sur ce fait, que ces travaux de transformation coûtent fort cher en temps et en efforts intellectuels, si bien que le développement de ce système nous apparaît irrémédiablement compromis tant et aussi longtemps qu'il ne pourra parler en « humain ordinaire », sans requérir que le juriste utilisateur ne doive s'habituer par surcroît à réfléchir et s'exprimer dans des schémas logiques trop complexes. Comme je reste convaincu que les problèmes les plus graves et les plus apparents trouveront néanmoins des réponses satisfaisantes dans un avenir assez rapproché, je ne me laisserai donc pas arrêter pour l'instant par ces difficultés, afin de passer à l'aspect sémantique du langage général du système de Meldman.

MAC, on s'en souviendra, ne peut se contenter d'*écouter*, il doit également *comprendre*. Voyons comment il prétend y arriver.

— *Le système entend*

Reprenant notre cheminement là où nous l'avons quitté, nous nous rappellerons que notre système avait pu percevoir nos messages formulés en PSL. Pour comprendre après avoir écouté, notre système analysera ensuite nos messages structurés sous forme de propositions grâce à un autre sous-système, ici nommé OWL, un logiciel destiné au traitement des modèles logiques (*a computer modeling system*), qui permet ultimement, après les traitements nécessaires de l'ordinateur, la comparaison et le calcul des propositions en cause. Laissant derrière nous les problèmes purement informatiques, arrêtons-nous plutôt à l'architecture logique du système, espérant voir raisonner MAC.

Comme nous l'avons soulevé déjà, le grand problème substantiel des systèmes juridiques intelligents se traduit dans le défi à relever de construire des modèles structurant les situations factuelles, et des systèmes permettant de marier ces modèles à d'autres, plus généraux, rapportant les réponses du droit s'appliquant à ses faits spécifiques, peu importe que ces normes émanent de la loi, de la doctrine ou de la jurisprudence.

Meldman, comme tous les autres, a rencontré ici ses pires difficultés, mais il prétend les avoir surmontées en empruntant des moyens qui nous rappelleront un peu les structures de JUDITH.

Le système de Jeffrey Meldman traite ainsi en plus des propositions factuelles catégorisées que nous connaissons déjà, d'autres propositions, juridiques en quelque sorte, qui sont de petits sous-ensembles normatifs structurés à partir d'une source juridique globale que l'auteur appelle le *Corpus Juris Mechanicum* (c.j.m.), lesquels serviront en principe de points de

retombée, lorsque les faits soumis (les *facts-at-hand*), le résultat des propositions de faits originales, seront insérées au modèle global, ou exemplifiées (*instantiated*) comme le décrit Meldman.

Les procédures exactes qui permettent cette exemplification appartiennent aux rouages internes des logiciels PSL et OWL de sorte que nous en ignorons les détails précis d'application. Il est permis de supposer toutefois que ces processus s'effectuent sur un plan autonome, de la façon que nous avons déjà décrite plus haut. Sur ce point, notre système manifesterait donc des signes d'intelligence, puisqu'il montrerait une action d'aspect raisonnable résultant du jeu de sa logique interne. Il reste cependant qu'il nous faut encore vérifier si les inférences qui sont ainsi réalisées par ce système découlent du fait que les notions qu'il a saisies et qu'il traite, avaient été vraiment *appries* ou *comprises* par lui, ou si elles découlent tout simplement d'un ingénieux traitement de concepts que notre système aurait par ailleurs justement analysés.

Partant de la conclusion que nous livre l'auteur lui-même au terme du texte qu'il a publié au *Rutgers Journal of Computers and Law*, peu de doutes subsistent¹³⁸. Ce système *entend*, mais il ne comprend pas. Il réussit à procéder par analogies uniquement parce que ces analogies éventuelles ont été définies comme telles par le requérant au système¹³⁹, avant même que ce dernier ne traite ensuite les propositions logiques qui lui sont soumises par rapport aux modèles préétablis qui permettent ses opérations logiques subséquentes. Notre système n'infère donc par lui-même que sur un plan très limité et pour une bonne part, parce qu'on lui a déjà « soufflé » un peu la réponse.

Même vu de loin, à travers un texte par ailleurs étoffé, ce système laisse transparaitre d'évidentes limites non seulement au point de vue de la préparation des données qu'il traite¹⁴⁰, et à celui de son accès général, mais aussi au plan de la constitution même de son logiciel, qui ne semble pas comporter d'instructions entraînant des opérations activement logiques. Toute son action paraît exclusivement construite sur des opérations de propositions, sans pouvoir engendrer, de lui-même, d'inférences qu'on ne lui aurait pas suggérées préalablement. Le compilateur dont on se sert ici manquerait donc de puissance interne, une carence qu'on trouvera peut-être corrigée au moins partiellement chez TAXMAN, dont nous aborderons l'étude, non sans avoir hautement apprécié toutefois les indéniables vertus de

138. *Supra*, note 135, à la p. 71.

139. Voir nos remarques, *supra*, note 137.

140. L'auteur souligne lui-même qu'il faudrait construire des milliers de modèles comme le sien, pour couvrir l'ensemble du droit. Peut-être est-il préférable que nous ignorions par ailleurs quel temps est nécessaire pour réaliser un tel système!

MAC. Il nous a en effet appris cette leçon précieuse voulant qu'un système intelligent doit arc-bouter son architecture sur les trois pièces maîtresses :

- d'un logiciel d'analyse grammaticale (*parsing program*)¹⁴¹ permettant à un tel système d'écouter des messages émis et de les recevoir, avant d'en retransmettre lui-même ;
- d'un logiciel de gérance de modèles (*computer modeling system*) traduisant la solide construction d'un processeur de logique juridique sans faille¹⁴² ; et,
- d'un langage spécial puissant (un compilateur), activant des macro-instructions qui commandent les opérations logiques les plus complètes possibles.

Comme il serait déraisonnable d'exiger de quiconque la perfection totale, nous ne viserons dès lors qu'à rechercher ici les systèmes les moins imparfaits, ou les plus satisfaisants, un don que nous a peut-être fait déjà le juriste L. Thorne McCarthy en réalisant son système TAXMAN, auquel nous allons maintenant porter toute notre attention.

2.1.2.4. TAXMAN

Comme nous l'avons fait dans le cas du système de Jeffrey Meldman, nous n'allons pas considérer TAXMAN¹⁴³ pour la question de fonds qu'il s'emploie à traiter, car elle soulève un problème de fiscalité corporative vue par le biais d'une très ancienne décision des tribunaux en la matière¹⁴⁴ sans intérêt pour nous. C'est pour son fonctionnement et les qualités de son logiciel que Taxman retient donc notre attention, et aussi allons-nous aborder son observation en commençant par en décrire les rouages internes.

— *L'univers de TAXMAN*

Ainsi que nous l'avons suggéré déjà, les systèmes d'intelligence artificielle structurent en cherchant à reproduire par des modèles analogiques aux

141. Arbib (*supra*, note 131, p. 317) définit cette opération d'analyse comme « the process of breaking a string of words into the pieces — such as noun phrase and verb phrase — specified by the syntax ».

142. Certains auteurs (v. g. Slayton, Coben ou Henkel) croient que le processus le plus important ici est celui des raisonnements de l'avocat. Nous croyons qu'il faut plutôt considérer ici en priorité la construction logique de la norme elle-même, puisque c'est elle qui conditionne finalement l'action du juriste, et non l'inverse !

143. L. T. MCCARTHY « Reflections on TAXMAN: An Experiment in Artificial Intelligence and Legal Reasoning », (1977) 90 *Harvard Law Review*, 837-893. Nous tirons nos renseignements sur TAXMAN de ce texte.

144. *United States v. Phillis*, 257 U.S. 156 (1921).

univers humains correspondants, des micro-univers artificiels où l'on parle des langages artificiels et où les agents qui l'habitent, ne conversent qu'à propos des éléments qui composent le champ clos de leur univers.

Dans tous les cas, la structure qui supporte ces mondes fabriqués de toutes pièces par un système créateur s'échafaude à partir de la technique des modèles et s'active grâce au processus des relations¹⁴⁵ établies entre les éléments qui s'y trouvent, ordonnées par un système logique aux fonctions prédéterminées.

Les systèmes d'intelligence artificielle génèrent donc deux ordres de problèmes complexes au moins, le premier, celui de la modélisation, cherchant à recréer des *patterns* analogiques de la réalité, et le second, pertinent à l'activation de ces modèles par la voie d'un logiciel processeur de traitement des interrelations entre les éléments du modèle constitué¹⁴⁶. Nous allons donc nous demander maintenant, d'abord comment TAXMAN reproduit analogiquement, ou *modelise*, la «réalité» de l'article 368 de l'*Interval Revenue Code* et, ensuite, de quelle façon il ordonne les relations entre les éléments de cette «réalité», et même de quelle façon il peut admettre de «nouveaux» concepts dans cet univers apparemment fermé. Si TAXMAN prétend à l'intelligence, il devrait en effet pouvoir *apprendre*, c'est-à-dire ajouter au répertoire de ses connaissances acquises, dont la somme totale ne devrait pas rester figée dans la structure des modèles à partir duquel des raisonnements devraient ensuite s'élaborer depuis le pré-acquis enregistré du système.

— Les modèles de TAXMAN

Comme ce fut le cas précédemment pour le système de Jeffrey Meldman, l'univers de TAXMAN se construit sur l'assemblage et la vérification subséquente de propositions logiques, exprimant des relations entre les éléments du micro-univers ainsi généré.

Chacune de ces propositions, qui établit en fait une relation entre un objet et un prédicat, constitue une *convention sémantique* ou en d'autres termes une *liste*¹⁴⁷, un énoncé qui prend place dans la banque de données du système.

145. Voici une notion capitale en intelligence artificielle ! L'informatique traditionnelle manipule des *choses* en rythme le plus souvent séquentiel ; elle travaille en *fichiers*. Ici, l'ordinateur traite les *relations* entre les choses en recherchant les adresses où les informations sont stockées ; la machine travaille en mode *listes* : SKYVINGTON, *supra*, note 39, p. 213.

146. Il existe effectivement un troisième problème possible soit celui de la traduction en langage naturel (ou du langage naturel). Nous le mettons de côté pour le moment puisque TAXMAN ne parle pas le langage naturel.

147. Voir *supra*, note 145.

Lorsque deux listes ont des prédicats communs, elles peuvent être réunies par paires, auxquelles il est possible d'ajouter encore d'autres propositions reliées, générant ainsi des ensembles de faits structurés (*cluster of facts*), ou des réseaux sémantiques (*semantic networks*). Ces réseaux constituent l'unité active fondamentale du système. Celui-ci devient par ailleurs complet par l'ensemble de sa banque de données (*data base*) constitué en dernière analyse de toutes les propositions contenues en mémoire du système.

Prise en elle-même, l'architecture de TAXMAN ne se montre pas révélatrice de l'essence même de ce système, dont l'âme véritable s'identifie au langage de traitement qui active ses analyses conceptuelles et sémantiques¹⁴⁸. C'est dans cette direction que nous allons donc maintenant orienter notre observation.

— Les déductions de TAXMAN

Comme nous l'avons dit déjà, on parle normalement dans un micro-univers artificiel, une langue aussi artificielle que l'interlocuteur externe doit également employer, à moins qu'il ne puisse compter sur les services d'un traducteur qui lui permette de s'exprimer au départ dans une forme de langage naturelle. Chez TAXMAN, comme précédemment avec MAC, un tel traducteur n'existe pas, de sorte que programmeur et requérant devront par conséquent parler la langue du robot, soit le MICRO-PLANNER¹⁴⁹.

L'utilisation qu'on fera de TAXMAN sera donc d'autant plus enrichissante, qu'on pourra maîtriser les *commandes* ou instructions de ce langage dont le lexique et la syntaxe sont plutôt complexes, mais par ailleurs assez puissants pour mettre en œuvre des processus qui aboutiront ultimement à générer des formes primaires de raisonnement. Dirigeons-nous vers ce résultat en signalant les instructions principales de PLANNER, en suivant les étapes fondamentales de son fonctionnement.

148. Avertissons immédiatement notre lecteur qu'il doit tenir compte de l'étrange paradoxe qui l'oblige à se méfier ici de la signification du terme *sémantique*. Dans l'environnement de PLANNER, une *sémantique*, résultat d'une relation établie entre les choses, traduit une *procédure*, et non une *signification* comme c'est le cas habituellement (Cf : G. MOUNIN, *Clefs pour la sémantique*, Paris, Seghers, 1972, p. 8 : également du même auteur, *Clefs pour la linguistique*, *id.*, 1971, p. 133). Ici, *meanings are procedures* ! « Les mots ne désignent pas directement les "choses", mais plutôt la manière de les exploiter — Reconnaître un objet, c'est savoir s'en servir ». (SKYVINGTON, *supra*, note 39, p. 291 et nos remarques *infra*, note 151).

149. Le MICRO-PLANNER est un dérivé de PLANNER, un langage de haut niveau conçu pour les travaux en intelligence artificielle : PLANNER provient lui-même d'un autre langage créé aux mêmes fins, soit le langage LISP.

En première phase d'opération, TAXMAN devrait d'abord pouvoir enregistrer en mémoire les conventions sémantiques à partir desquelles il élaborera ensuite ses raisonnements subséquents, suivant ce que lui dicteront à cet effet les macro-instructions que lui signifiera l'interlocuteur requérant. Cette entrée en mémoire est ici rendue possible par l'instruction ASSERT qui adresse des listes en mémoire de l'ordinateur.

Naturellement incomplète par elle-même, l'opération d'entreposage doit se compléter d'une opération de repérage, essentielle lorsque le système raisonnera sur un ensemble (*cluster*), dont il devra évaluer tous les éléments, en vérifiant entre autres choses la *présence* des propositions soumises dans la banque de données. Cette opération est rendue possible par l'instruction GOAL. Une recherche négative à ce point du raisonnement mettrait fin ici en principe à toute l'opération, mais une réponse positive entraîne au contraire des développements au sein de l'ensemble du réseau où le requérant travaille.

En faisant appel à une autre instruction, appelée PROG, il est en effet possible de réussir une première espèce de déduction, découlant par exemple du traitement séquentiel de quatre propositions qui constitueraient les éléments constitutifs d'une autre proposition, laquelle ne serait pas contenue à la banque de données.

Une telle opération n'a rien de compliqué pour *Homo sapiens*, qui va aisément conclure qu'une proposition est vraie, si tous les éléments nécessaires pour qu'elle existe se trouvent réalisés en même temps. On sait par exemple qu'un contrat est valide, si on réalise que les quatre conditions qu'imposent les dispositions de l'article 984 du *Code civil*, sont remplies en même temps à propos d'un contrat donné.

Si on réalisait en PLANNER, à partir d'une hypothèse originale donnée, une application précise de cet article du *Code civil*, chacun des énoncés/conditions de ce texte constituerait une *liste* dont il faudrait vérifier la présence par rapport à d'autres listes déjà emmagasinées dans la banque de données et dont les termes contenus dans ces propositions seraient déjà eux-mêmes définis, dans cette même mémoire. Une fois ces vérifications faites et les réponses affirmatives obtenues à quatre reprises, la proposition/hypothèse originale serait par conséquent vérifiée et déclarée correcte. C'est là une opération simple apparemment pour l'homme, mais complexe pour l'ordinateur et son programmeur, dont le rôle ici est absolument essentiel, puisque le raisonnement est en définitive le sien, beaucoup plus que celui du système, à qui on a soufflé, sinon la réponse, du moins la façon de s'y prendre! TAXMAN serait-il donc aussi incapable de réussir des analogies sur un plan autonome? Faut-il vraiment à chaque requête lui réindiquer la définition de tous les *termes* et de toutes leurs interrelations possibles pour

qu'il puisse compléter son raisonnement? Ne peut-on pas compter en d'autres mots, sur une macro-instruction GOAL surpuissante, qui active pour le requérant en même temps une instruction PROG incluse, qui réalise une déduction produisant la reconnaissance d'un concept, et qui fasse en sorte que le système garde ce résultat en mémoire pour terminer ultimement son raisonnement à l'aide de ce nouvel élément, cette variable définie par le système lui-même?

S'il existe une réponse à cette question, on la retrouvera à partir du système TAXMAN lui-même sans doute, mais également, et même peut-être mieux encore dans PLANNER, vers lequel nous nous tournons donc maintenant.

— Les inférences de PLANNER

De conception fort différente des logiciels traditionnels, PLANNER est orienté prioritairement en fonction du traitement des relations entre les objets beaucoup plus qu'en fonction des choses elles-mêmes comme c'est le cas normalement. PLANNER, dit-on, est « *goal-oriented* »¹⁵⁰. La sémantique des propositions traduit pour lui non pas de simples significations, mais plutôt des tâches à accomplir, dont il définit le déroulement par la reconnaissance qu'il tire de sa mémoire, des propriétés des objets qui meublent son univers. Porté à juger des réalités par les prédicats des choses, il participe dès lors presque naturellement au processus analogique qui caractérise l'intelligence par l'une de ses facettes au moins.

Pour permettre à PLANNER d'atteindre ses objectifs, son créateur, Carl Hewitt¹⁵¹, l'a doté de structures actives appropriées dont l'une se nomme fort justement THEOREM et dont la fonction consiste à « suggérer » au système global pourquoi il devrait considérer applicable un prédicat donné à une variable soumise, étant donné que cette relation a été établie déjà dans d'autres conventions connues où la réunion d'éléments similaires produisait des résultats identiques à ceux que l'on espère maintenant.

150. Voir: SKYVINGTON, *supra*, note 39, p. 290.

151. PLANNER est le logiciel moteur du robot Shrdlu de Ted Winograd. Hewitt, constatant que la logique mathématique ne se révélait pas capable de résoudre seule les difficultés du *problem-solving*, conçoit de remplacer l'idée de *vérité logique*, par celle de *procédure*, où le robot connaît les objets non par leur définition traditionnelle, mais par la façon par laquelle la chose arrive à être ce qu'elle est. Comme le dit Skyvington, « la machine propose en quelque sorte, non seulement le « quoi », mais aussi le « comment » (SKYVINGTON, p. 290). PLANNER connaît les choses « par la procédure qu'il faut suivre pour les fabriquer ». Dans cette théorie dite « d'épistémologie procédurière », la sémantique réfère donc bel et bien à une procédure, plutôt qu'à une signification (voir mes remarques, *supra*, note 148, et surtout l'interview de Carl Hewitt, rapportée par SKYVINGTON, *supra*, note 39, pp. 283 et sq.).

Si on désirait ainsi amener un TAXMAN civiliste à juger de la validité d'un contrat conclu entre X et Y, on se servirait de THEOREM pour rappeler au système global *comment*, en d'autres circonstances, on avait atteint cette conclusion par suite de l'étude de certaines listes pertinentes à un tel problème. On l'amènerait ainsi à remarquer en même temps que la procédure par laquelle cette validité s'était construite autrefois lors de notre connaissance de l'entente X-Y, se retrouve aujourd'hui par rapport à la convention A-B, par exemple. De là découle l'inévitable conclusion.

On notera que le concept de *validité* n'est pas ici considéré *in se* par ses éléments constitutifs, mais plutôt par les procédures¹⁵² qui mènent à la conclusion que la validité existe ou non. Le concept de *validité* ne se traduit pas dans une chose, mais sous forme d'une propriété¹⁵³ constatée pertinente à un, ou, par extension¹⁵⁴, à plusieurs objets donnés.

En étant capable de justifier l'application de prédicats donnés à des variables, TAXMAN, par l'entremise de PLANNER, se révèle donc capable d'analogies autonomes, et témoigne ainsi d'une forme au moins rudimentaire de raisonnement applicable à un problème juridique. Selon McCarthy, TAXMAN pourrait saisir la description d'un cas de réorganisation de compagnie au niveau approximatif de l'opinion légale et pouvoir de là produire une analyse située à un degré d'abstraction suffisant pour atteindre la conclusion attendue¹⁵⁵.

Malgré l'admiration justifiée que suscite cette réussite sur le plan informatique, il n'est sans doute pas inutile de se demander maintenant si le résultat qu'on obtient d'un tel système, justifie la somme d'efforts nécessaires pour compléter un simple syllogisme que le juriste ordinaire résout par ses propres moyens de façon beaucoup plus économique. Les limitations de TAXMAN sautent aux yeux! L'implantation d'un tel système coûte fort cher et nécessite un temps de préparation fort considérable. Or, comme il faut constituer une banque de données étendue pour en rendre l'usage intéressant, les difficultés se multiplient d'autant. Que dire enfin de l'accès au

152. Nous rappelons encore nos remarques, *supra*, notes 148 et 151.

153. Gardons ici la définition de *propriétés* que suggère Jolley en ce sens qu'elles « sont les attributs des objets ». (J. L. JOLLEY, *Le traitement des informations*, Paris, Hachette, 1971, p. 14).

154. Le *Théorème de conséquence* peut travailler de deux façons suivant qu'on lui demande d'agir en mode ABSTRACT ou EXPAND. Dans le premier cas, une requête est menée à un niveau d'abstraction supérieur, alors que les recherches se conduisent en même temps de façon automatique à un niveau inférieur; on dit alors: (GOAL < modèle conceptuel > ABSTRACT). Dans le deuxième cas, on soumet une question de fait à un niveau supérieur et, automatiquement, le programme génère les niveaux inférieurs; on dit alors: (ASSERT < modèle conceptuel > EXPAND). Voir: L.T. MCCARTHY, *supra*, note 143, p. 861.

155. *Id.*, p. 864.

système, fort pénible, difficile en raison de la complexité de PLANNER, un langage puissant peut-être sous maints aspects, mais aussi lent et inefficace, suivant les termes mêmes du professeur McCarthy¹⁵⁶ !

Les systèmes intelligents constituent certes le prolongement naturel de nos systèmes traditionnels d'informatique juridique¹⁵⁷, mais il ne faut pas compter les voir opérationnels et accessibles sur une échelle acceptable et rentable d'ici plusieurs années, soit quand ils pourront répondre à certains critères d'acceptabilité, dont nous allons esquisser brièvement les traits principaux, au terme de cette étude, dont les conclusions équivaldront en fait à formuler des projets.

2.2. La logique de l'apprentissage

Plusieurs savants semblent pouvoir décrire un système intelligent. Il devrait, dit-on, être capable de percevoir et d'apprendre, être motivé, pouvoir suivre une démarche intentionnelle et résoudre des problèmes, jouir de la faculté de penser et de symboliser¹⁵⁸. C'est là dresser un bien ambitieux programme pour cet élève peu doué qu'est l'ordinateur auquel on s'accorde généralement à attribuer un quotient intellectuel équivalant à 10.

D'après ce que nous avons vu précédemment, JUDITH, MAC et TAXMAN arrivaient d'ailleurs à au moins simuler déjà plusieurs des performances que nous venons de signaler, et nous avons conclu pourtant qu'ils n'étaient pas très brillants. Peut-être l'intelligence ne s'observe-t-elle que par rapport à une échelle graduée. Un chroniqueur italien a fait remarquer fort justement « que les machines sont incapables de faire preuve d'intelligence, mais les êtres humains non plus »¹⁵⁹.

De crainte dès lors de calomnier des robots de bonne souche en les jugeant moins malins qu'ils ne le sont en réalité, ou par crainte de louer à l'excès des hommes de science qui seraient plus automates qu'ils ne se l'imaginent, gardons notre mission finale plus modeste, en essayant de rappeler les conditions essentielles du système artificiel pensant et en élaborant les très grands traits principaux d'une recherche originale sur ce sujet.

156. *Id.*, p. 876.

157. Nous avons cité plusieurs auteurs déjà qui ont soutenus cette thèse dont, *inter alia*, MCCARTHY, *id.*, 888 ; MELDMAN, *supra*, note 135, p. 40 ; POPP and SCHLINK, *supra*, note 125, p. 305 ; BUCHANAN and HEADRICK, *supra*, note 123, pp. 41 et *sq.* ; P. SLAYTON, *supra*, note 83, pp. 29-30 ; BOYD, *supra*, note 92, pp. 268 et *sq.*

158. Ces conditions sont énoncées par le cybernéticien F. H. GEORGE in « Intelligence et cybernétique », n° spécial de la revue *Science et Vie, Le cerveau et la pensée*, p. 92, à la page 97.

159. *Id.*

2.2.1. Les conditions du système

Un système artificiel prétendant à la pensée devrait au départ faire la démonstration d'au moins quatre facultés intellectuelles pour être reconnu raisonnable par *Homo sapiens*, et témoigner au surplus d'une facilité d'accès telle qu'il puisse se révéler un interlocuteur efficace et rentable pour son créateur.

Notre légiste automate devrait donc pouvoir nous démontrer qu'il peut converser et communiquer, apprendre et inférer, le tout dans un cadre agréable et efficace. Mais voyons comment se traduisent ces caractères.

— Être conversationnel

On n'imagine pas qu'un système intelligent puisse fonctionner en différé. On attend de lui qu'il réponde directement à nos interventions et modifie ensuite ses comportements en conséquence.

Autrement, il ne pourrait faire la preuve qu'il peut structurer des messages, c'est-à-dire démontrer qu'il pense en maîtrisant un langage. Or, puisque notre robot doit satisfaire au paradigme qui le condamne à ressembler à l'humain, il doit par conséquent être capable de s'exprimer, de la même façon que le fait l'homme lui-même. « If there is any one capability that is uniquely human », nous dit Wooldridge, « it is the power of speech »¹⁶⁰.

Le langage de notre système sera ainsi articulé sur des liaisons logiques autonomes, et non précodées automatiquement, car il ne pourrait être alors intentionnel et il deviendrait incapable d'apprentissage en raison de son prédéterminisme rigide. Or, un système intelligent devrait être communiquant en mode bi-directionnel, c'est-à-dire en émettant et en assimilant des concepts, si on veut toujours garder notre paradigme intact¹⁶¹.

— Être communicant

Notre système se trouve dès lors obligé de mener un dialogue authentique avec son interlocuteur.

On se souvient qu'Eliza¹⁶² se révélait à ce chapitre une brillante

160. D. E. WOOLDRIDGE, *The Machinery of the Brain*, New York, McGraw-Hill, 1963, p. 153.

161. Il faut distinguer ici le système *conversationnel* du système simplement *interactif* où l'intervention de l'ordinateur est générée par le programme, automatiquement ou suite à une commande activée par l'utilisateur du système. Dans le système PLATON, par exemple, il suffit de presser au clavier la touche NEXT pour entraîner le système à réagir par la production d'un nouveau message.

162. Voir SKYVINGTON, *supra*, note 39, p. 176.

simulatrice, en retournant à son interlocuteur des phrases que le système assemblait mécaniquement par les rouages de son logiciel limité à la reconnaissance de signes formels. L'ordinateur ne formulait donc pas de réponses véritables à l'autre partie puisqu'il n'était pas capable de structurer ni la réception ni l'émission en rétroaction de messages sémantiques.

L'analyse sémantique est donc essentielle à la constitution du système intelligent. Sans elle, le dialogue ne peut jamais être formateur de concepts nouveaux. Or, notre système devrait aussi :

- être capable d'apprentissage conscient, et,
- être capable d'inférence créatrice.

Le phénomène de l'apprentissage se traduit essentiellement par l'assimilation de notions nouvelles, enregistrées par le sujet de l'acte d'apprendre, en vue de rendre ce dernier capable de corriger les erreurs de son acquis préalable et de prévenir et empêcher qu'il construise par la suite des raisonnements erronés en structurant des messages à partir des données de ce préacquis.

Comme on le voit, l'acte d'apprentissage manifeste clairement qu'il émane d'une machine essentiellement cybernétique¹⁶³ où le processus de rétroaction¹⁶⁴ remplit à merveille son double rôle de correcteur de faute et de catalyseur d'enseignant de la norme juste. Bien des systèmes corrigent automatiquement leurs malfunctions. Les bibliothèques de logiciels informatiques regorgent de tels programmes. Les démarches que ces appareils génèrent restent cependant totalement mécaniques ; on dirait, en termes juridiques, que l'ordinateur ne garde ici qu'une compétence absolument liée.

Pour notre robot intelligent, nous devons donc préconiser la création d'un logiciel qui puisse lui permettre un apprentissage conscient¹⁶⁵ dont il ait par conséquent un contrôle générant des décisions qui émanent du seul fonctionnement de sa propre logique. Nous souhaiterions que notre système obéisse moins à des réflexes qu'à des choix intentionnels. De là dépend le fait que notre système idéal puisse raisonner et créer par ses propres facultés.

163. Voir nos commentaires dans « La machine et le droit... », *supra*, note 11, pp. 492-493.

D'après L.-J. DELPECH, une machine cybernétique est une « machine qui ne travaille plus selon des programmes rigides mais qui peut s'adapter, dans certaines conditions, à des situations non prévues par son constructeur » : « Un créateur : Norbert Wiener », in *Le dossier de la cybernétique*, Verviers, Éd. Gérard et Co., 1968, p. 37.

164. N. WIENER définit le *feedback* ainsi : « The property of being able to adjust future conduct by past performance », in *The Human Use of Human Beings*, New York, Avon Books, 1950, p. 47.

165. RASTRIGUINE et GRAVÉ définissent le conscient comme une « [forme de régulation permettant] de gouverner dans une conjoncture inhabituelle le comportement d'un système... dont le contrôle dans les situations courantes est assuré au niveau du subconscient ». (in : *La cybernétique telle qu'elle est*, Moscou, MIR, 1979, p. 41).

Dans l'état des connaissances actuelles, nos cerveaux électroniques sont nantis de tronc cérébraux ¹⁶⁶ suffisamment élaborés pour leur permettre des automatismes assez impressionnants. Certains logiciels leur fournissent même matière à occuper la première couche d'un cortex rudimentaire, mais il n'en demeure pas moins qu'on n'a pas dépassé encore ici les premières ébauches du robot pensant rentable et efficace.

Pour que cette branche spéciale de l'informatique qui expérimente la conception d'intelligences dites artificielles parvienne à une maturité véritable, elle devra produire en effet des systèmes dont l'utilité à court terme s'impose auprès de la clientèle moyenne qu'on désirait servir lors de leur création. Suivant la cinquième condition que nous avons annoncée, notre système doit :

— *Être efficace et rentable*

À cette époque où le *friendly system* devient l'inévitable règle ¹⁶⁷, il est inacceptable en effet que de tels systèmes restent l'apanage exclusif de quelques spécialistes qui sont assez savants pour pouvoir contrôler les complexités de langages hermétiques et aux constructions logiques inhabituelles. De tels systèmes sont voués au rejet systématique et c'est pourquoi JUDITH, MAC et TAXMAN ne paraissent pas pouvoir prétendre à une exploitation enrichissante pour de longues années à venir.

Nous rencontrons ici un problème ardu de logiciel, d'autant plus épineux qu'il manifeste nos difficultés à contrôler les mécanismes du langage ordinaire, et que nous ne savons que fort peu de chose sur le raisonnement humain, surtout quand il est au surplus juriste ! Un très vaste programme de recherches s'ouvre donc devant nous, et c'est en en traçant les lignes maîtresses, que nous souhaitons conclure notre étude.

2.2.2. Le projet de système

Suivant ici la voie que nous indiquait plus haut Jeffrey Meldman ¹⁶⁸, il nous faudra développer nos efforts suivant les trois axes qui nous sont dessinés par les besoins structuraux essentiels du système intelligent. On devra ainsi concevoir :

166. Le tronc cérébral, qui origine de la moelle épinière pour se développer généralement par le mésencéphale, est le « vieux » cerveau responsable chez l'homme des fonctions vitales de son existence, telle la respiration, ou qui traite les messages primaires, comme la faim et la soif.

167. Voir nos remarques en première partie à propos des systèmes sympathiques.

168. Voir nos remarques, *supra*.

- a) Des logiciels d'analyse grammaticale opérationnels permettant à notre système d'*écouter* des messages et d'en transmettre;
- b) Des logiciels générateurs et gérants de modèles logiques; et
- c) Un compilateur puissant permettant l'opération d'un système efficace et d'accès abordable par tous les intéressés.

Sous plusieurs aspects, on pourra songer à spécifier quelques-uns de ces instruments. Certains les voudront de culture française¹⁶⁹ et d'autres préféreraient les créer juristes, encore qu'on pourrait tout simplement tenter de les créer comme juristes de culture francophone, une mission qui relèverait fort bien de l'intérêt de légistes intéressés au progrès et au développement de leur difficile profession.

Pour enivrante que l'idée puisse être, sans doute faut-il en aborder néanmoins la réalisation avec réalisme et prudence, étant donné la très grande complexité des problèmes que soulève un tel défi. Pour atteindre les objectifs globaux qu'on pourra se fixer, il semble bien qu'on doive procéder par étapes, depuis les préacquis sûrs actuels, jusqu'aux hypothèses les plus extraordinaires ensuite.

D'après ce que nous avons observé déjà, et tenant pour acquis qu'une coopération internationale suivie est indispensable pour atteindre le succès final, il me semble qu'on devrait dès lors :

— d'abord privilégier le développement et l'exploitation de l'informatique juridique traditionnelle, puisque les techniques qu'elle emploie semblent avoir atteint un plateau de stabilité et de maturité qui témoigne de la justesse des solutions qu'elle propose¹⁷⁰.

Doit-on ajouter au surplus que la preuve de son utilité apparaît aujourd'hui bien établie, surtout en matière de traitement des lois, où le succès des systèmes qu'on a implantés devrait convaincre les plus sceptiques. Comme nous l'avons démontré plus haut, les sujets de recherche ne manquent pas ici encore, particulièrement au point des

169. Comme on l'a vu déjà, ABF, MAC et TAXMAN nous semblent bien représenter des systèmes adaptés aux *Common Lawyers*. Il faudrait se méfier de l'assimilation sournoise que pourrait souffrir le juriste francophone par le biais du langage et de la philosophie substantielle qui sous-tendrait un logiciel de conception anglo-saxonne. À force de penser comme un *Common Lawyer*, peut-on devenir *Common Lawyer*?

170. Il est évident que nous prenons sur ce point précis le contre-pied des conclusions exposées déjà par Philip Slayton, mais *Res ipsa loquitur* ! L'informatique juridique a d'ores et déjà démontré qu'elle procurait des avantages considérables à court terme. Pourquoi dès lors en empêcher le développement ? Par contre, le problème de l'intelligence artificielle soulève des difficultés telles, que même les centres les plus prestigieux qui traitent de ce problème (v.g. MIT, Stanford etc...) n'ont pas encore construit de systèmes vraiment opérationnels sur ce sujet. Doit-on bloquer le court terme au profit d'un long terme dont la définition reste au surplus incertaine !

phénomènes généraux de synonymies et d'équivalences sémantiques de même que sur le plan de l'acceptation des systèmes, loin d'être réussie encore.

- Par ailleurs, il ne faudrait pas négliger non plus l'informatique de l'an 2000, celle des systèmes intelligents, le prolongement naturel de celle que Skyvington appelle déjà « l'informatique de papa »¹⁷¹.

Conclusion

Nous venons d'élaborer les grandes lignes de la recherche qui s'annonce ici, et nous ne reviendrons pas dès lors sur ce sujet, si ce n'est pour ajouter une dernière touche à ce grand tableau d'ensemble, pour y esquisser la dernière dimension de tout ce phénomène, soit celle qui définit l'effet culturel de l'information électronique sur le monde juridique.

L'ordinateur, médium électronique par excellence, extension de ce système nerveux¹⁷² dont dépendent nos processus affectifs et cognitifs, va-t-il modifier, même imperceptiblement, sur une longue période de temps, notre conception du droit et notre façon d'appliquer les normes légales ?

Si la mutation de nos lois imprimées s'accélérait, notre droit écrit pourrait bien redevenir verbal¹⁷³. D'élite qu'il est présentement, il deviendrait éventuellement plus convivial¹⁷⁴, car les transformations par le droit informatisé atteindront les structures mêmes du monde juridique.

Sur un plan plus rapproché, et plus réaliste peut-être, devrait-on se préparer à la décentralisation administrative que prévoient Simon Nora et Alain Minc¹⁷⁵ ? Or, plus radicalement encore, une pratique du droit conçue dans les cadres que dessine Alvin Toffler¹⁷⁶, reconstituerait à long terme un

171. SKYVINGTON, *supra*, note 39, p. 213.

172. M. McLuhan, *The Medium is the Massage*, New York, Bantam Book, 1967, pp. 40-41.

173. Restons « McLuhanien » ! Lorsque le droit écrit devient électronique, il change de nature en se modifiant comme médium. De médium *chaud*, il devient médium *froid*. Le médium chaud se définit ainsi : « Medium (with much detail or definition) being supplied by the transmitter, leaving the receptor little to add from his own experience or knowledge » (H.H. Crosby and G.R. Bond, in « Introduction », *The McLuhan Explosion*, New York, American Book, 1968, p. 7).

174. « Une société conviviale est une société qui donne à l'homme la possibilité d'exercer l'action la plus autonome et la plus créative, à l'aide d'outils moins contrôlables par autrui ». (I. Illich, *La convivialité*, Paris, Seuil, 1973, p. 43).

175. S. Nora et A. Minc, *L'informatisation de la société*, Paris, La documentation française, 1978 ; les lecteurs « pressés » en trouveront un résumé par F. Harrois-Monin dans « Serons-nous demain les "sauvages" d'un monde informatique ? », dans le numéro d'août 1978 de la revue *Science et Vie*, p. 74.

176. A. Toffler, *La troisième vague*, Paris, Denoël, 1980 ; les lecteurs « pressés » en trouveront un résumé dans « La maison électronique », *L'Express*, le 7 juin 1980, p. 74.

monde juridique individualisé, de nouveau activé par de petits bureaux électroniquement reliés par terminal, télévision bi-directionnelle ou autrement, à de grands ensembles informatifs gérant de riches banques de données.

Ces mutations pourraient bien sûr se briser sur d'irrésistibles oppositions aux changements, fréquentes en milieu juridique. Malgré ses réticences premières, notre profession finit néanmoins toujours par s'adapter aux modifications de son environnement. N'est-elle pas d'ailleurs l'une des plus vieilles du monde? Sans doute, face à l'informatique, se laissera-t-elle encore une fois porter par la nouvelle vague, ... la *troisième* naturellement !