

Les Nouvelles d'APL

Revue Trimestrielle d'AFAPL

(Association Francophone pour la promotion du langage APL)

N°25 Décembre 1997

Dans ce numéro :

- Éditorial	B. Mailhol
- Réflexions sur l'intégrale de parité de Langlet et ses applications	M. Locquin
- Ce siècle avait 99 ans...	B. Mailhol
- Place d'APL dans le monde actuel et à venir	M. Dumontier
- Roles of APL in Satellite Surveillance avec le résumé de cet article en français par M. Dumontier	Jack G. Rudolph
- APL, une approche matricielle pour la détermination des taux zéros-coupons	H. Ralison
- La quadrature du cercle ? ...c'est « possible » : plus précis que le palmer!	M. Dumontier
- Comment j'ai commencé à travailler..., m'amuser avec APL	B. Secq
- Les concepts généraux des I*Net	B. Mailhol

Et les rubriques régulières

- Manifestations: APL98 à Rome	P. Di Chio
- Guide des Produits et Services APL	AFAPL
- Courrier des lecteurs	S. Gajdoczki A. Niervèze M. Dumontier

AFAPL

Association Francophone pour la promotion du langage APL

Association régie par la loi de 1901
174 bd. de Charonne - 75020 PARIS - FRANCE
Tél. & Fax: (33)-01-43-56-31-79
<http://www.ensmp.fr/~scherer/langlet/>

Constitution du bureau

Président M. Bernard Mailhol
Vice-Présidents M. Sylvain Baron
M. Eric Lescasse
Trésorier M. Bernard Secret
Trésorier adjoint M. Raymond Tisserand
Secrétaire G^{al} Mme Ludmila Lemagnen
Secrétaire adj. M. Patrick Paumard

Revue « Les Nouvelles d'APL »

Imprimée par AFAPL
Dépôt légal: juin 1996
Numéro ISSN: 1664-4699
Directeur de la Publication:
M. Bernard Mailhol

Sociétés membres (1997) : **Dassault Electronique** (J.L. Crottier), **DynaSys** (F. Schott), **Banque de France** (H. Sinturel), **BM** (B. Mailhol), **Sylicom** (S. Baron), **Uniware** (E. Lescasse), **Legrand Consultants** (B. Legrand), **Ministère de la Culture** (J.M. Rouzou), **Thomson** (Madame P. Pons).

Les noms entre parenthèses sont ceux des correspondants.

Comment devenir Membre d'AFAPL

Faire parvenir votre Bulletin d'Adhésion (voir à la fin de ce numéro) rempli, accompagné d'un chèque bancaire couvrant la cotisation annuelle au Secrétariat ou directement au Trésorier

Mme Lemagnen	M. B. Secret
Secrétaire G ^{al} d'AFAPL	Trésorier d'AFAPL
174 bd. de Charonne	35 r. Jules Ferry
75020 PARIS	91200 ATHIS-MONS

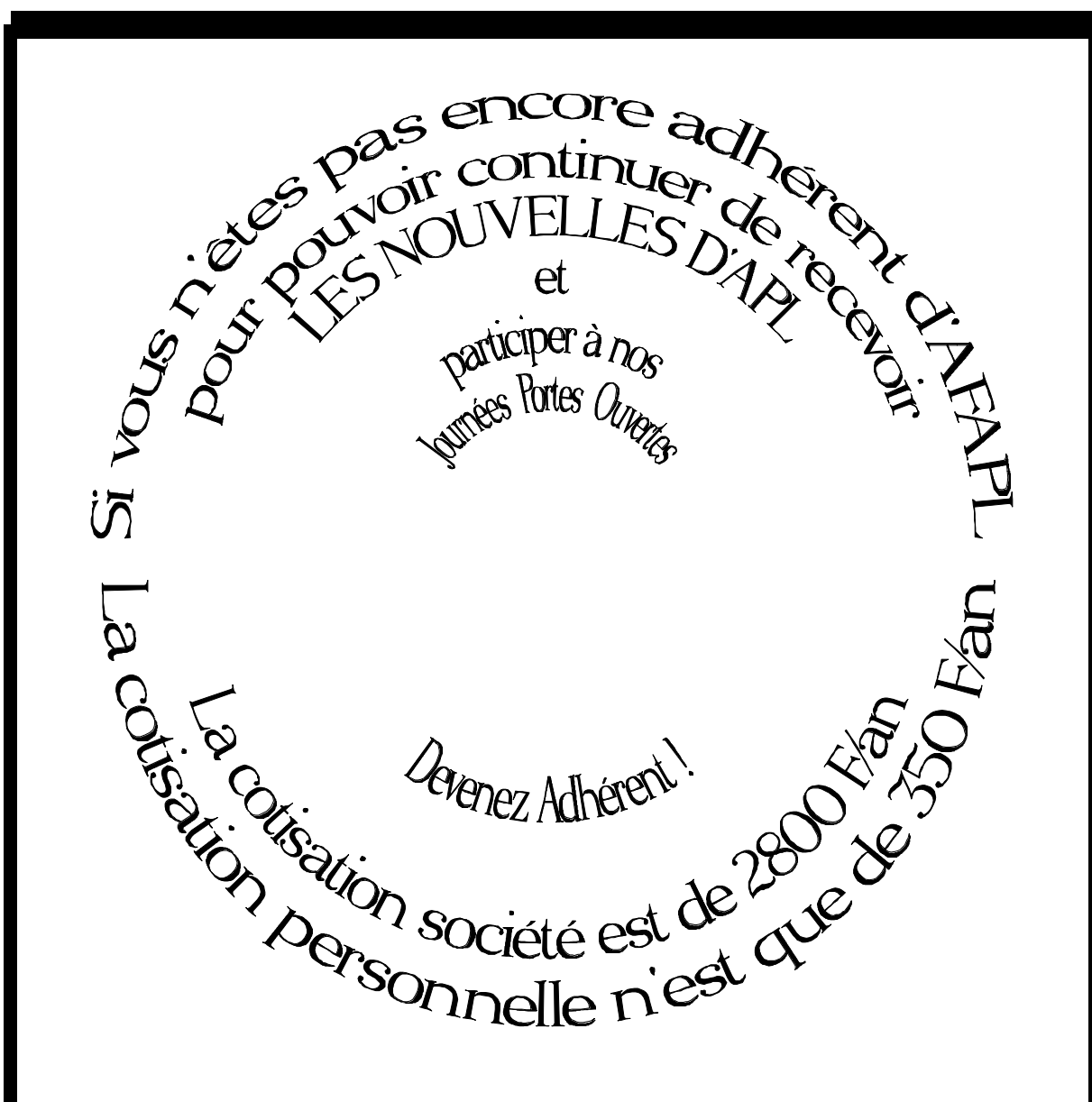
Cotisation	Montant
Personnelle	350 F
Société	2800 F

Ajouter 100F pour un abonnement à l'étranger. La cotisation **Société** donne droit à figurer gratuitement dans le Catalogue des Produits et Services APL.

Tarif des publicités (voir : Calendrier des prochaines parutions)

Sommaire

- Éditorial du Président	B. Mailhol 3
- Note de la Rédaction	 5
- La Vie de l'Association et les parutions prochaines	 6
- Réflexions sur l'intégrale de parité de Langlet et ses applications	Marcel Locquin 7
- Ce siècle avait 99 ans...	B. Mailhol 31
- Place d'APL dans le monde actuel et à venir	M. Dumontier 51
<u>Actualité internationale</u>		
- Roles of APL in Satellite Surveillance avec le résumé de cet article en français par Michel Dumontier	Jack G. Rudolph 59
- APL, une approche matricielle pour la détermination des taux zéro-coupons	H. Ralison 81
- La quadrature du cercle ?... c'est « possible » : plus précis que le palmer!	M. Dumontier 91
- Comment j'ai commencé à travailler..., m'amuser avec APL	B. Secre 97
- Les concepts généraux des I*Net	B. Mailhol101
<u>Manifestations à venir</u>		
APL98 à Rome	Paolo Di Chio127
- Les joyeusetés du correcteur orthographique de Word	La Rédaction130
- Guide des Produits et Services APL	AFAPL131
- Courrier des Lecteurs	S. Gajdoczki 141
	A. Niervèze 143
	M. Dumontier 145
- Bulletin d'Adhésion et de Contact	 147



La cotisation personnelle donne droit à

- recevoir un exemplaire des Nouvelles d'APL (par trimestre)
- participer aux ½ journées Portes Ouvertes
- adresser un CV pour publication, en cas de recherche d'emploi

La cotisation Société donne droit à

- recevoir 5 exemplaires des Nouvelles d'APL (par trimestre)
- être listé dans notre «Guide des Produits et Services APL»
- pouvoir inclure des Publicités (1000 F par page, 600 F par ½ page)
- participer aux ½ journées Portes Ouvertes

Éditorial du Président

par Bernard Mailhol

Voici le numéro 25 des nouvelles de l'APL. Il est partiellement le reflet de notre dernière journée "Portes ouvertes"; il évoque aussi l'évolution de l'usage de ce langage, de son impact sur nos applications.

Tout d'abord, Marcel Locquin évoque quelques uns de ses travaux menés avec Gérard Langlet, depuis "Factotum" , jusqu'à la prise de conscience de l'importance de "l'intégrale de parité", et ses applications innombrables, dont la liste - je l'espère - va continuer à s'étendre au fil des prochaines années ou décennies.

Nous apprenons aussi que les travaux de Gérard Langlet vont être accueillis dans un musée de Grenoble. Nous évoquerons plus longuement ce sujet dans un prochain numéro.

Ce journal porte aussi l'accent sur les usages de l'APL. Des usages opérationnels. De même que la pratique de l'APL a conduit Gérard Langlet à représenter - avec succès - son univers par des vecteurs booléens, la pratique de l'APL conduit à imaginer des applications industrielles que les contraintes habituelles de coût et de délai nous interdisent, lorsqu'elles sont développées avec d'autres outils, que ce soit dans le monde de la gestion (Article de H. Ralison), ou dans le monde mathématique de la Défense (article de Jack Rudd).

Ces articles répètent inlassablement que le développement d'applications en APL est possible, efficace, industriel.

Ils laissent aussi penser que l'APL ne sert que dans des applications très complexes, mais d'autres articles, montrant des bases de statistiques, mathématiques ... laissent au contraire penser qu'APL ne traite que d'applications très simples, mais ne peut servir à développer des 'vraies' applications.

L'article de Bernard Secret forme une réponse à ces hésitations : il évoque la mise en place d'applications répondant à des besoins généraux de gestion des entreprises : dans ce domaine - concurrencé par les tableurs - APL apporte la capacité de développer des applications pouvant garantir la qualité d'une donnée, être (parfois) documentées, puis maintenues.

Enfin, un article évoque les concepts généraux des I*Net. Cet article ne concerne pas directement APL, mais fait partie de la culture de base que doit aujourd'hui avoir tout développeur. Il concerne quand même APL, car nous allons développer - en APL - aussi bien des "serveurs Web", que des "Clients Web" particuliers. Avant que les APL contiennent des outils chargés d'organiser ces travaux, vous pouvez les développer à peu de frais (à peu de minutes) en connaissant quelques concepts de base. Vous les trouverez dans cet article. Des explications complémentaires sur l'écriture elle-même des clients et des serveurs se trouvent dans le numéro 23.

Tout ceci revient à dire : faites de l'APL, c'est un langage important, bien qu'en-dehors de la mode. Son usage ne peut que vous permettre d'avoir de meilleures idées, d'oser imaginer des applications directement utiles à vos travaux, oser les mettre en place, ne serait-ce que pour prouver que ces applications sont réalisables ... car réalisées.

Ainsi, vous pratiquez APL, et souhaitez continuer à faire partie de l'AFAPL. C'est le moment de penser à votre inscription pour 1998.

De même, nous vous rappelons que le congrès APL 1998 a lieu à Rome, du 27 au 31 juillet 1998. Si vous souhaitez présenter une communication, vous pouvez présenter un premier résumé au 15 janvier prochain. Vous trouverez toutes les informations dans ce numéro des Nouvelles, ou en interrogeant le serveur du Congrès en

<http://www.economia.utoverm.it/apl98>. (Ce site est en construction).

Note de la Rédaction

Il est rappelé que les articles (ou les lettres à reproduire) doivent être accompagnés d'une disquette. Les fichiers doivent s'imprimer en format A4 (le format de la revue est réduit en A5 seulement à la reproduction).

Ne pas tenter d'incorporer des styles particuliers pour les titres et sous-titres, car cela accroît la tâche de mise en page homogène du numéro dans son ensemble. (Éviter d'utiliser des fichiers de type .DOT).

Utiliser en général la police « Times New Roman » .TTF en corps 14 (notes et remarques en corps 12, jamais en dessous).

Lorsque des polices spéciales sont nécessaires (APL ou autres), il est recommandé, soit de fournir ces polices soit d'utiliser l'option de sauvegarde des fichiers avec leurs polices (option de Word6 « Enregistrement » du Menu « Outils » : « Incorporer les polices True Type »).

La Vie de l'Association

Calendrier des Prochaines Parutions

	Articles	Publicités	Parution
N° 26	15 février 1998	28 février 1998	mars 1998
N° 27	15 mai 1998	28 mai 1998	juin 1998
N° 28	15 août 1998	28 août 1998	septembre 1998

Les dates indiquées sont les **dates-limite de remise des articles et des publicités**. Envoyez-nous vos articles sous forme de fichier texte pur ou, de préférence, sous forme de document Word. (On peut maintenant accepter des articles écrits sur Macintosh (Word 4.1 ou 5.1) aussi bien que sur PC.) Dans tous les cas, nous souhaitons recevoir aussi une impression contrastée, type laser, de vos articles. (Éviter les envois de textes sans disquette, merci). Voir également les recommandations de la Note de la Rédaction.

Tarif des publicités

Les tarifs des publicités sont les suivants

Type	Tarif HT
1 page complète	1000 F
1/2 page	600 F

Les typons doivent être remis, accompagnés du règlement, au plus tard un mois avant la parution du numéro dans lequel ils doivent être insérés.

Les publicités peuvent nous parvenir sous forme de pages A4 imprimées en noir et blanc sur papier blanc, de préférence sur imprimante laser. Elles seront reproduites telles quelles dans la revue de notre Association.

Réflexions sur l'intégrale de parité de Langlet et ses applications

par Marcel V. Locquin

Résumé

Nous exposons dans ce qui suit les travaux effectués avec Gérard Langlet entre fin 1982 et début 1996. Essentiellement concentrés sur le traitement du langage descriptif des objets biologiques et techniques en langues naturelles, ils ont abouti à la rédaction de 3 600 programmes écrits en APL, de 300 tableaux de données accessibles en ligne, le tout engrangé sur une seule disquette formaté DOS à 900k, et de quelque 300 bases de données biologiques et linguistiques disponibles séparément. Nous terminons par des réflexions sur les applications de l'intégrale de parité qui nous paraît être une découverte fondamentale transdisciplinaire d'application aussi universelle que la transformée de Fourier.

Note importante:

Les textes «entre guillemets et en italique» sont des citations textuelles de Gérard Langlet datant de 1995, en réponse aux textes que je lui avait soumis, en prévision d'un livre que j'avais alors en préparation sur les applications physiques et biologiques de l'intégrale de parité. Se jugeant incapable - ce mot dur est de lui - de rédiger un texte accessible au public non logicien, il avait préféré m'en confier la rédaction, en en acceptant la révision bien entendu.

Mots clés

ADN, algorithme, APL, asymétrie, biologie, Boole, C, chaos, code, cogniton, créativité, cristallographie, décimal, dessin, domino, fortran, génétique, graphisme, hélicon, informatique, intégrale, intelligence, interférences, langage, Langlet, langues, linguistique, logique, LPA, médecine, morphogénèse, optique, parité, pariton, physique, propagation, psychologie, quanton, relativité, rythmes.

Introduction

Domino était souvent le premier à m'accueillir à Jouy-en-Josas lorsque je rendais visite à Jacqueline et Gérard Langlet dans leur demeure péri-forestière. La fonction domino d'APL avait été vitalisée par ses maîtres en la personne d'un adorable chat roux, maître des lieux et surtout du jardin, voire des bois alentour lorsqu'il prenait la tête d'une assemblée nocturne quasi vénitienne des chats du

voisinage. La puissance du domino d'APL à nulle autre pareille dans tous les autres langages informatiques alors connus, méritait bien me disait Gérard cet honneur insigne d'être en quelque sorte trans-vitalisé dans ce matou, de surcroît surnommé par son entourage "minouchet".

Je dois au regretté général Jean Becam, fondateur de l'A.I.L.F., Association des Informaticiens de Langue Française, le bonheur d'avoir connu et travaillé avec ce virtuose de l'informatique Gérard Langlet.

C'est en Décembre 1982, que j'ai demandé à brûle pourpoint à Jean Becam *"connaissez-vous un informaticien qui soit bien au courant du traitement informatique des langues naturelles ? "* Oui me répondit-il du tac au tac, sans aucun doute le plus fort c'est Gérard Langlet alors membre de l'AFCEC comme vous. Immédiatement je pris contact avec lui lors d'une réunion du Collège de Systémique de cette association qui était alors composée à 50% d'informaticiens. Elle vient hélas de disparaître presque en même temps que Gérard Langlet.

Réciproquement enthousiasmés par les perspectives ouvertes par nos premières conversations sur les langues, nous avons aussitôt décidé de nous lancer dans la grande aventure du traitement informatique des langues naturelles, non dans la perspective de traductions plus ou moins automatisées, mais dans la perspective de la taxinomie, c'est à dire du traitement du langage descriptif des objets techniques comme des êtres vivants. En ce domaine, nous étions parfaitement convergents car tous les deux physicochimistes, et complémentaires, Gérard étant plus informaticien que moi, moi plus biologiste que lui.

Dans ce dernier domaine, l'amour commun de la mycologie et de la mycogastronomie nous réunissait et nous participions chaque année aux journées mycologiques de Bellême dans l'Orne. Il y avait entre nous, une intercompréhension de physique fondamentale importante, puisque Gérard était entre autres un cristallographe hors pair et que de mon côté j'avais, dans les années 1950-1964, approfondi moi-même en physique l'optique et l'électronique interférentielles, en travaillant avec deux prix Nobels, le hollandais Fritz Zernike, inventeur du contraste de phase et le hongrois Dennis Gabor, inventeur de l'holographie, ce qui nous réunissait dans la connaissance notamment des techniques optiques d'analyse des structures cristallines ou para-cristallines, en lumière polarisée. C'est à cette époque que j'ai invité à Paris le cristallographe de renom, Ralph W. G. Wyckoff, pour présider le Premier Congrès International de Microscopie électronique dont j'étais l'organisateur.

Pour débiter dans nos travaux, Gérard et moi avons alors acquis un ordinateur portable "Ampère" de fabrication japonaise, qui avait pour particularité d'avoir le

langage APL gravé dans sa puce, ce qui libérait toute sa mémoire vive pour travailler, à une époque où la capacité des mémoires vives était fort maigre. Dès qu'ATARI sortit ses premiers modèles basés sur le microprocesseur 68 000 de Motorola, nous avons pensé aussitôt qu'y charger l'APL 68 000 nous permettrait de commencer à travailler de façon plus extensive. Nous avons alors utilisé, en parallèle, chacun un MégaST 4, avec les mêmes configurations et les mêmes accessoires, pour pouvoir, entre deux séances de travail, coordonner nos travaux par téléphone si nécessaire.

Gérard me répétait souvent deux vérités dont j'étais déjà convaincu, «*l'avenir de l'informatique passe par le traitement et la programmation en langue naturelle, sans codage préalable par l'opérateur*» et «*la nature ne fait pas de calculs, elle n'utilise que des fonctions logiques*».

La première vérité n'a que rarement sauté aux yeux des informaticiens. En effet, dès les débuts de l'informatique, ses développements ont été confiés aux mathématiciens qui en avaient besoin pour améliorer les performances de leurs calculs. Le fait, par exemple que l'addition ne soit pas une opération naturellement simple, à la portée des êtres vivants autres que l'homme, a été rarement reconnu. En effet peu d'informaticiens programmeurs sont au courant des "*hiérarchies enchevêtrées des algorithmes biologiques*", cette expression est de Douglas Hofstadter. Les généticiens préfèrent la simplifier arbitrairement en la nommant "*code génétique des programmes biologiques*".

J'utilise volontairement le qualificatif très fort de l'informaticien Douglas Hofstadter qui me paraît particulièrement adéquat depuis l'effondrement du dogme central de la continuité génétique dû aux travaux de deux prix Nobels, la britannique Barbara Macintosh qui a mis en évidence les "gènes sauteurs" des chromosomes du Maïs et le japonais Tonagawa "les gènes fluents" qui glissent le long de la double hélice de l'ADN des chromosomes humains.

Règles de comportement

Avant le début de notre travail commun, nous nous étions donnés cinq contraintes opérationnelles fondamentales:

1.- Un chercheur doit pouvoir travailler et publier dans la langue de son choix. Il est aussi absurde de vouloir imposer l'anglais à un francophone ou un germanophone, que le français à un japonophone ou un anglophone. Les outils que nous allions développer devaient donc être translinguistiques.

2.- Ce choix étant fait, il faut faciliter l'autonomie linguistique des chercheurs géodécentralisée. Un chercheur formé dans une spécialité, à Bruxelles, à Montréal ou à Paris, doit pouvoir continuer à se perfectionner de façon autonome une fois qu'il est sur le terrain, même dans un pays non francophone et sans possibilités d'accès à des fonds documentaires dans sa spécialité.

3.- Les commandes au clavier de son ordinateur doivent pouvoir être faites directement en langue française ou dans sa langue maternelle, seules des abréviations logiques immédiatement interprétables pouvant être acceptées.

4.- Tous les programmes devaient pouvoir tenir sur une seule disquette formatée DOS à 900k, afin de faciliter l'autonomie et les échanges économiques par la poste.

5.- Un éventail de caractères typographiques, hors l'alphabet romain, devait pouvoir être directement disponible au clavier.

Le LPA

Gérard Langlet, avant notre rencontre, avait initialement développé un langage pédagogique audiovisuel en langue française, le LPA, acronyme palindromique d'APL. Suivant son conseil, je me suis procuré une cassette contenant le LPA, auprès du licencié de l'époque par le CEA, la société SOFREMI qui a disparu peu après.

Très vite, en dialoguant avec Gérard, nous sommes tombés d'accord sur le fait qu'il fallait étendre le champ d'action du LPA à autre chose que le calcul ou la programmation.

MYCOLOC

C'est pour cela, qu'après la disparition de la SOFREMI, nous avons progressivement remplacé le LPA par un nouvel outil, étendu au traitement des langues naturelles, nommé INTERPRETE et incorporé au sein de MYCOLOC. Ce faisant nous l'avons fait évoluer afin qu'il devienne un outil taxinomique général, c'est à dire capable de nous permettre de travailler sur des descriptions d'objets techniques ou biologiques rédigées en toutes langues naturelles, sans codage préalable, et ce, en alphabets multiples, latin, cyrillique, hiragana, katagana et avec un certain nombre d'hiéroglyphes égyptiens. A tout moment dans MYCOLOC on peut l'appeler en tapant la commande INTERPRETE. En cours de travail, sa présence est signalée par un point discret en début de chaque ligne. On en sort très simplement en tapant au clavier le mot FIN.

Sous INTERPRETE on peut par exemple taper la demande suivante:

"donne le carré de l'inverse de la racine cubique de pi multiplié par le nombre d'or à la puissance quatre", ou "calcule les indices taxinomiques", ou "donne le centratype", sous-entendu de la base en cours de travail.

Muni de ces outils je les ai à ce moment utilisés sur une trentaine de bases construites pour la détermination des genres et des espèces de champignons. Gérard s'est alors amusé à réaliser un programme en APL pour dessiner à l'écran un objet de révolution, comme un champignon à chapeau, calculé à partir de sa demi coupe tracée sur le bord gauche de l'écran à la souris.

FACTOTUM

C'est à partir de cette époque que des fonctions graphiques se sont développées dans FACTOTUM devenu le successeur enrichi de MYCOLOC. Certaines images réalisées par Gérard étaient pour moi fort inattendues comme le plan du Métro de Paris et du RER, avec toutes ses stations et l'optimisation puis la simulation du parcours à partir des deux stations de départ et d'arrivée pointées à la souris. En fin de travail la durée du parcours et les stations où devaient s'effectuer les correspondances s'affichaient à l'écran. Ou encore par application d'une méthode originale d'optimisation des graphes eulériens le plan de situation de 36 villes de France, de 48 villes britanniques, puis de 101 villes allemandes. La commande pour optimiser le parcours du voyageur de commerce entre les villes françaises est ITINERAIRE puis VASY, entre les villes britanniques CITIES n puis VASY, entre les villes allemandes STAEDTE 101 suivi de GEHE. En même temps, s'affichait le parcours graphique, la longueur du parcours final en kilomètres ainsi que le temps mis pour le calculer.

Dès le début du développement de FACTOTUM nous avons convenu de numéroter chaque version dès que nous avons considéré qu'il fallait faire une pause dans le développement, pour passer à des applications effectives de la version considérée. C'est ainsi que le dernier, produit en commun, est FACTOTUM 233. Il contient, sur une seule disquette, 3 600 programmes écrits en APL-II et 300 tableaux de données contenant notamment les 2 300 mots les plus courants en langue japonaise translittérés en caractères latins, que l'on peut reconvertir en hiragana ou katagana par les outils de translittération contenus dans FACTOTUM.

Cette performance a été atteinte grâce à ce que Gérard nommait la «*compression algorithmique*», basée essentiellement sur les performances extrêmes des

fonctions d'APL, ainsi que sur une logique nouvelle concentrée de programmation et d'articulation des programmes par concaténation. Ceci a permis d'aboutir à un programme général de traitement des données naturelles intitulé par Gérard lui-même «*propagation asymétrique de la parité*» ou encore plus simplement «*intégrale de parité*» dont je parlerai longuement plus loin.

Une nouvelle logique de programmation

Lorsqu'il a intuitivement senti que la compression algorithmique conduisait à une nouvelle logique de programmation universellement applicable à toutes les structures naturelles, Gérard a fait preuve d'une prudence exemplaire. Il m'a simplement dit un jour, «*j'ai découvert quelque chose qui bouleverse notre compréhension de la programmation biologique, mais je ne t'en parle pas encore en détails car je veux auparavant être sûr que je ne suis pas fou, tellement c'est simple*».

Il a ensuite attendu un an pour m'en expliciter la logique binaire simplissime en APL. Pendant cette année de méditation algorithmique, nous avons continué, en parallèle, à travailler sur les langues, ainsi que dans bien d'autres domaines. Gérard me disait souvent «*en plus d'APL j'ai plusieurs outils secrets qui me permettent de travailler très vite*».

De ces outils, j'en ai connu au moins deux, le premier est la transposition semi-automatique en APL de programmes écrits en FORTRAN, le second est lié aux programmes écrits en C. Il me demandait souvent de lui signaler des programmes écrits en C entrant dans notre champ de recherches, pour pouvoir disait-il, sans les réécrire, les transcrire en APL dans FACTOTUM. S'il a parlé en clair de son utilisation du FORTRAN, ce qu'il a fait avec le C m'a été très obscur. Il me disait souvent, «*je ne souhaite pas que FACTOTUM soit commercialisé en son entier car il contient trop de choses et il ajoutait, «si un jour on commercialise des morceaux de FACTOTUM, il faudra les réécrire en C puisque c'est ce langage qui est le plus universellement utilisé pour travailler sur les ordinateurs PC, mais il n'est pas performant pour le développement comme l'est l'APL*». Il ajoutait, «*réécrire en C, un programme en APL, pour moi ce n'est pas un problème bien que je ne sache pas encore bien utiliser ce langage*».

Autre outil semi-secret pour la programmation, grâce à la fonction "domino" de l'APL, il obtenait la conversion des matrices en tableaux enregistrés comme du texte pour économiser l'espace mémoire, avec la possibilité de les restituer automatiquement en matrices lors de leur appel dans un programme.

Les conjugaisons

La conjugaison de tous les temps de tous les verbes de la langue française, y compris les irréguliers, les défectifs et les pronominaux, ainsi que ceux qui se conjuguent avec des verbes auxiliaires comme faire, est appelée par la commande: CONJUGUE' suivie du verbe français à l'infinitif.

En espéranto, la conjugaison passé, présent, futur, est appelée par la commande ESPERANTO puis effectuée par ESP' verbe-français. Les conjugaisons latines y sont appelée par LAT' suivie de l'infinitif latin, comme par exemple LAT'amare.

Les déclinaisons latines

La commande est DECLINE' suivie de la phrase latine complète telle que DECLINE' *hic homo sapiens erectus*, ou DECLINE' *genus tricholoma, species anceps, forma insolita*.

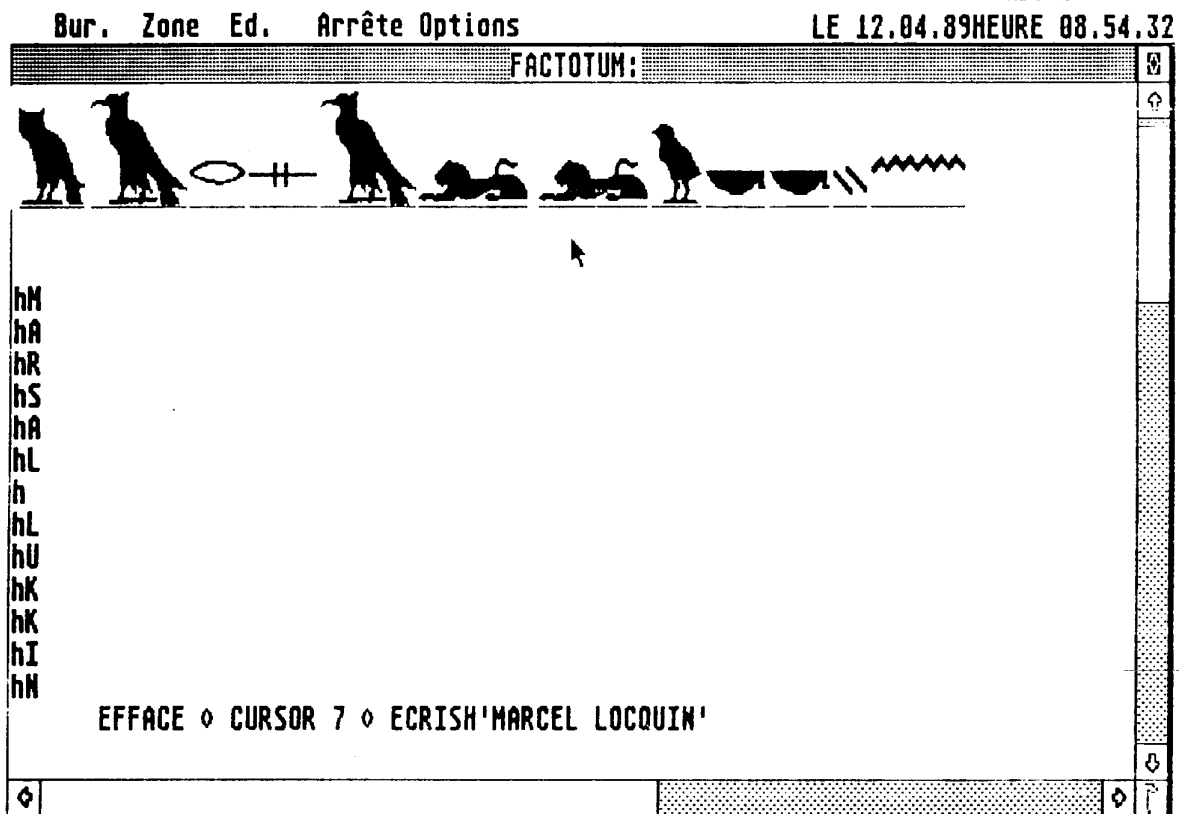
Les translittérations

On appelle l'alphabet slave par CYRILLIQUE, puis on effectue la translittération par la commande RUSSE' suivie des mots français comme par exemple RUSSE' yeux noirs.

On appelle l'alphabet grec par la commande < grecque > en minuscules, suivie de l'instruction < grec' > toujours en minuscules, comme par exemple: grec' Hippocrate.

Pour le japonais on charge la police katagana par la commande SCAN puis en frappant REVELE'KATA, ou REVELE'HIRA, suivi de JEU'KATA ou HIRA, on affiche à l'écran l'alphabet japonais katagana ou hiragana. La commande ECRISJAP' mot, le translittère en katagana. et 1ECRISJAP' mot, le translittère en hiragana. J SCRABBLE'IMO signale la présence de la chaîne IMO dans l'ensemble J.

Pour charger les hiéroglyphes égyptiens la commande est IMH. Pour translittérer mon nom en hiéroglyphes par exemple, la commande est ECRISH'Marcel Locquin, fig. 1.



Applications médicales

Dans FACTOTUM il y a des outils d'applications médicales, verrouillés car, me semble-t-il, effectués à la demande du CEA. Il s'agit d'ANALMED et de MEDICIS, acronyme de "Medical Intelligent Service". ANALMED est un outil classique de gestion des résultats d'analyses biologiques de laboratoire, qui n'appelle de ma part aucun commentaire.

MEDICIS est beaucoup plus original et ses applications débordent la médecine pour s'étendre à toutes les sciences diagnostiques, au premier rang desquelles se trouve la taxinomie. MEDICIS intègre les expériences diagnostiques des utilisateurs éventuellement interconnectés en réseau, issues des observations engrangées par eux en pondérant automatiquement leurs critères de décision.

La présence de chaque critère diagnostique est notée 1 par l'opérateur, son absence 0, ce qui crée une ligne diagnostique constituée de 0 et de 1. L'ensemble des diagnostics sur un même thème construit une matrice sur laquelle s'applique automatiquement le programme en fin de session. La commande DIAGNOSTIC affiche la pondération des critères de -3 000 à +3 000. La réactualisation est faite automatiquement en fin de chaque session. L'utilisateur de MEDICIS est fort bien guidé en cours de travail par des instructions en ligne. A ma connaissance on a utilisé MEDICIS pour des diagnostics en médecine, en mycologie, en

psychologie, en psychiatrie, en détection des pannes, en analyse du discours politique,

Fonctions taxinomiques

Les fonctions taxinomiques de MYCOLOC sont trop nombreuses pour que nous les citions toutes. Voici les principales:

APPLICATION crée une nouvelle application - MODIFIE n permet de modifier le taxon n - TAXONS' donne la liste des taxons - TAXON 1 affiche le taxon 1 - DERNIER affiche le dernier taxon - RETASSE élimine les taxons vides à la fin - INDEX' donne la liste des descripteurs - LOIGNEMENT DE n classe par ordre décroissant de parentés à partir du taxon n tous les taxons de la base - PARENTS DE n donne les taxons les plus proches du taxon n - CROISE'aaa&bbb&ccc donne les descripteurs aaa, bbb, ccc, simultanément présents dans tous les taxons extraits préalablement par PARENTS DE n - TX commande le calcul automatique de l'ensemble des indices taxinomiques: coefficients de similarité de Jaccard, indice d'affinité de Sokal et Michener, rapport de similarités de Rogers et Tanimoto, coefficient d'association d'Hamann, ainsi que la fiabilité du résultat selon Locquin, la pertinence des résultats selon Locquin, ainsi que la validité globale des résultats d'après Locquin - CENTRATYPE donne le taxon central de l'ensemble des taxons - APOS donne les coefficients d'apomorphies des taxons - CLASSIFIE'mycota permet de présenter hiérarchiquement par menus déroulants la classification de la base "mycota" - SUCCESEURS'genre-p et PREDECESEURS'espèce-s affichent le ou les taxons de rang immédiatement inférieur à p ou supérieur à s - en cas d'erreurs classificatoires, la frappe de PROBLEMES permet de vérifier et éventuellement de rétablir la cohérence de l'ensemble.

MYCOLOC est maintenant incluse dans FACTOTUM. Il contient environ une cinquantaine d'autres commandes qui permettent un travail taxinomique, classificatoire notamment, facile et approfondi sur tous les textes décrivant des objets techniques ou biologiques, descriptions écrites en toutes langues naturelles utilisant les alphabets latin, grec, cyrillique, hiragana, katagana, voire hiéroglyphique.

La version incluse dans FACTOTUM contient FLEUVES qui engrange l'ensemble des fleuves français et leurs affluents qui sont appelés par des instructions comme LOIRE AFFLUENTS ou TRIBUTAIRES par exemple.

Intermède interprète

A tous moments, pendant le travail taxinomique, on peut appeler INTERPRETE et procéder à des calculs, des déclinaisons, des conjugaisons, des translittérations, à l'écriture de sous-programmes. On peut également commander des fonctions graphiques ou appeler des graphismes déjà engrangés.

Graphismes

Avec la commande DESSINE'AMANITES, ou toute autre image préalablement construite et engrangée, comme FLECHE, ABEILLE, LEPIOTES, BOLET, STAEDTE, CHAMPIGNONS, MYCENES, CAMPANULOIDE, LEVROIDE, OMPHALOPHALLUS, BRETAGNE, CORSE, VILLES, VASE, TONNEAU, MOTIFS, on obtient l'affichage à l'écran de leurs images.

Avec la commande IMAGE, IMAGE GROSSIE ou IMAGE SURGROSSIE ou IMAGE RENVERSEE ou IMAGE SYMETRIQUE DE LA FLECHE, on modifie en conséquence une image comme celle de la flèche.

On peut combiner ces fonctions et demander IMAGE SURGROSSIE RENVERSÉE CADRÉE DE MOTIF, si MOTIF est le nom d'une image préalablement construite et engrangée dans la base.

IMAGEF 0 2 4 6 8 10 révèle les images paires de la base en ligne jusqu'à la dixième.

TORE'CYCLO crée un tore dont la section droite est CYCLO.

Logique relationnelle

A côté des fonctions logiques il existe 7 fonctions relationnelles dont 2 sont également logiques, ce sont: inférieur à - inférieur ou égal à - supérieur à - supérieur ou égal à - appartenant à - égal à (logique) - différent de (logique). Toutes les fonctions relationnelles peuvent être définies à partir d'une combinatoire de fonctions logiques.

Il y a en outre deux autres fonctions logiques spéciales appelées "opérateurs", car on peut les appliquer, en une seule opération, à chaque item d'un vecteur, autrement dit d'une chaîne de nombres, ce sont:

- réduction ou compression, notée / (barre oblique)
- propagation ou expansion ou cumul, notée \ (rétrobarre).

Examinons de plus près ces deux opérateurs:

On place avant l'opérateur réduction l'une des fonctions arithmétiques + - : * < > que l'on veut appliquer sur tous les items: +/ est la réduction additive (lorsque appliquée à un vecteur qui est une séquence de bits elle donne le nombre de 1 dans la séquence) (lorsque appliquée à une séquence de nombres entiers elle donne la somme arithmétique de ces nombres).

+/1 2 3 4 donne 10.

-/ est la réduction soustractive.

*/ est la réduction multiplicative ou réduction de "fois", qui donne le dernier terme de la multiplication de chaque item par le résultat précédent:

*/1 2 3 4 donne 24

>/ est le plus grand item

</ est le plus petit item

On définit très simplement à la suite: l'étendue, les fonctions retiens et rejette par exemple.

Le symbole rétrobarre \ appelé opérateur expansion, ou propagation, ou cumul, peut être précédé des symboles arithmétiques + ou * et des symboles logiques = ou != différent.

La propagation de l'addition +\ fait la somme de tous les composants du vecteur:

+\1 2 3 4 donne 1 3 6 10

La propagation de "fois" donne la suite de termes multipliés par le précédent:

*\1 2 3 4 donne 1 2 6 24

Appliquons la propagation puis la réduction de la différence != à un vecteur binaire:

!=\0 1 1 0 0 0 0 1 0 1 1 1 0 0 0 0 0 1 1 0 1 1 0 0 donne

0 1 0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 0

=/= / 0 1 1 0 0 0 0 1 0 1 1 1 0 0 0 0 0 1 1 0 1 1 0 0 donne

0 c'est à dire le dernier terme du précédent .

Nommons une séquence binaire "B", avec les briques élémentaires: 0 1 et =/=, nous pouvons construire, en lisant les opérations dans l'ordre de leur exécution en APL, c'est à dire de la droite vers la gauche:

0 =/= B qui est la réplique à l'identique,

1 =/= B qui est le NON, la négation binaire,

B =/= B qui est la négation de soi-même, c'est à dire, la platitude, la mort,

1 =/= B =/= B qui est, après la mort, le remplissage compact de l'espace par des bits élémentaires, le passage à l'uniformité,

0 =/= B =/= B qui est, après la mort, le vide total de l'espace, le retour à l'incréd.

Gérard me fait remarquer: «*Attention au 0: il est à la fois l'anti-1 et l'absence d'information, donc aussi le zéro des mathématiciens. Dans une séquence telle que: 0 0 1 0 1 0 0, les deux 0 de gauche ne sont rien, alors que les autres sont des anti-1. Cette subtilité existe aussi en mathématique classique: un billet de 00100 francs vaut 100 francs*».

La parité

C'est une égalité de valeurs exprimée en binaire.

Qu'eut écrit Newton s'il avait connu Boole? Il eut sans doute inventé le calcul différentiel et intégral d'une autre manière. Au lieu d'utiliser le système décimal, il eut utilisé le système binaire booléen et inventé l'intégration binaire dont les propriétés sont extrêmement étonnantes.

On aurait ainsi pu économiser deux cents ans de développement mathématique arithmétique décimal, engendrant à l'ère informatique de multiples et grossières erreurs notamment produites par la propagation des troncatures du *n*ème chiffre significatif après la virgule aux nombres entiers avant la virgule, induite par les millions de répétitions par seconde effectuées par les ordinateurs actuels. C'est ainsi que se génère, à l'insu des opérateurs, ce que Gérard nommait «*le chaos*»

informatique». Il ajoutait «*ceux qui prétendent ainsi simuler sur ordinateur la théorie du chaos déterministe, ne font que développer le chaos informatique*».

La découverte de l'intégrale de parité, faite par Gérard Langlet, bouleverse et rénove entièrement notre façon de raisonner, aussi bien dans l'appréhension des sciences physiques, biologiques et humaines, que sociologiques et psychologiques, bref dans tous les domaines des sciences et des applications de ces sciences.

Pour affirmer qu'un processus physique ou biologique distingue la droite de la gauche, il est nécessaire de rattacher nos propres éléments de symétrie à ceux du système lui-même, car un modèle construit par le chercheur doit implicitement, sinon explicitement, adopter une convention d'orientation par rapport aux coordonnées fixées à priori.

C'est ce que Einstein appelait le "mollusque de référence", voulant dire par là que ce référentiel n'avait aucune existence réelle, ni intrinsèque, ni absolue. Nous sommes libres de le choisir et d'adopter n'importe quel système de référence, car la notion de droite et de gauche est entièrement subjective, elle n'existe que par rapport à notre propre référentiel corporel asymétrique.

La propagation de la parité est le mécanisme essentiel de toute évolution. G. Langlet l'a très bien montré.

Considérons une séquence d'informations B exprimée en binaire, par exemple:

0 1 1 0 0 0 0 1 0 1 1 1 0 0 0 0 0 1 1 0 1 1 0 0

La parité d'une telle séquence est 0 si le nombre de 1 est pair et 1 s'il est impair.

Dans le cas précité B est 0 puisqu'il contient dix fois 1. Autrement dit la parité est la somme de tous les items de B modulo 2, ou encore sa parité est le reste de la division de la somme de tous les items binaires de B par 2.

La parité est de ce fait très généralement utilisée lors de la transmission d'informations pour vérifier la qualité de la transmission. Pour ce faire on découpe le message en paquets égaux de bits et on insère en fin de chacun d'eux un bit qui exprime la parité de chaque paquet et qui est transmis avec lui. A la réception, le receveur recalcule les parités de chaque paquet, en fait la somme en fin de transmission et vérifie son identité avec la parité transmise. Cette procédure est une somme de contrôles modulo 2. S'il n'en est pas ainsi le receveur-ordinateur redemande une nouvelle transmission à

l'émetteur.

En fait, au lieu d'additionner les 1 et de calculer la parité cumulée des séquences, il est plus simple de propager seulement les bits successifs de parité en utilisant le ou-exclusif ou XOR, comme l'a démontré Gérard Langlet.

«Attention: Braitenberg se trompe s'il écrit: "Il existe deux autres relations ayant cette propriété: NI et OU EXCLUSIF". Les seules deux relations permettant de fabriquer toutes les autres sont, en logique: NI et NON_ET (NAND) (Peirce, Sheffer), mais... NON tout seul suffit en réalité s'il est conditionnel: IF A THEN B:=NOT B. Les logiciens avaient l'habitude de voir les équivalences en mode statique, ce qui n'a aucun sens physique: le IF sur A est nécessairement antérieur au basculement de B. Alors, les portes ET, OU_INCLUSIF, NI et NON_ET n'ont plus ni nécessité ni signification. Et OU_EXCLUSIF propagé remplace, de manière dynamique très économique, une kyrielle de portes ET normalement exigées dans un circuit statique: l'algorithme du cliquet de Locquin n'a aucun sens en statique, n'est-ce pas?».

Dans la notation de K. Iverson toutes les expressions sont simples: Le "ou exclusif" ou XOR ou "différent de", est noté \neq . On en déduit que:

$0 \neq 0$ est 0 (0 différent de 0 est faux)

$0 \neq 1$ est 1 (0 différent de 1 est vrai)

$1 \neq 0$ est 1 (1 différent de 0 est vrai)

$1 \neq 1$ est 0 (1 différent de 1 est faux)

les opérations XOR sont donc commutatives.

Le contraire de XOR est NON-XOR noté $==$ équivalent (à ne pas confondre avec le signe égal).

$0 == 0$ est 1 (0 équivalent de 0 est vrai)

$1 == 1$ est 1 (1 équivalent de 1 est vrai)

Notons en annexe que cette logique n'a pas besoin de l'axiomatique de Péano pour fonctionner. Il n'existe pour la logique booléenne selon Gérard Langlet qu'un seul axiome dit "axiome de parité", ainsi formulé:

«tout nombre n a un successeur de parité inverse et zéro n'est pas un nombre mais un signe».

Gérard poursuit:

«Il faut se méfier du mot "égalité", car l'égalité logique est la non-différence logique; en outre, deux négations équivalent à une affirmation: pour toute suite impaire de parités, on a un nombre pair de signes dans la chaîne; alors, l'égalité logique fait autant l'affaire que la différence logique, tandis qu'elle donne l'anti-résultat de celui de la différence pour tout terme pair:

$BBIN 'LOC' B: 010011000100111101000011$
 $=/\backslash B: 011101111000101001111101$
 $=\backslash B: 001000101101111100101000$ ».

Réflexion d'espace et parité

Une réflexion d'espace s'assimile à une inversion des coordonnées de l'espace associé au système. Elle permute la droite et la gauche. Une loi invariante dans cette réflexion est dite "conservatrice de la parité".

Les équations de Maxwell, l'interaction électromagnétique, entre autres conservent la parité.

Dans un modèle non relativiste, temps et espace étant strictement indépendants, la réflexion d'espace ne change pas ce qu'on appelle le sens de la flèche du temps.

Réflexion d'espace-temps et relativité

Dans un modèle relativiste par contre, la réflexion d'espace-temps figure une réflexion de l'univers quadri-dimensionnel. Il s'ensuit que l'on peut rechercher quels types d'interactions sont ou ne sont pas invariantes par rapport au temps et à la parité.

Si on ajoute au cas précédent la conjugaison de charge, les particules, qu'elles soient positives ou négatives, sont soumises aux mêmes lois électromagnétiques. Que l'on soit dans la physique pré-einsteinienne ou dans la physique relativiste, ou maintenant dans la physique quantique, aucun lien n'est évident entre les opérations paritaires et les opérations temporelles. Pour en découvrir un, il faut mettre en oeuvre le formalisme de la théorie quantique des champs, qui élargit la théorie uniparticulaire relativiste de Dirac. Dans ce modèle, à chaque particule correspond une antiparticule symétrique dont toutes les propriétés sont inversées: charge, impulsion, spin ... On parle alors d'invariance par rapport à C.P.T., charge, parité, temps. Pendant longtemps on a accepté l'idée que cette invariance était une loi de la nature.

Remarque de Gérard: «*Attention à la néguentropie... La Propagation Asymétrique de la Parité est isentropique*».

L'intégrale de parité

On peut formuler, selon Gérard un «*Axiome d'Hérédité: chaque parité "nouvelle" est la résultante de chaque parité "ancienne", cumulée à TOUTES les parités "nouvelles" de TOUS les prédécesseurs, indépendamment de l'ordre de ceux-ci. (une parité "nouvelle" du prédécesseur est "ancienne" pour la parité courante...)*».

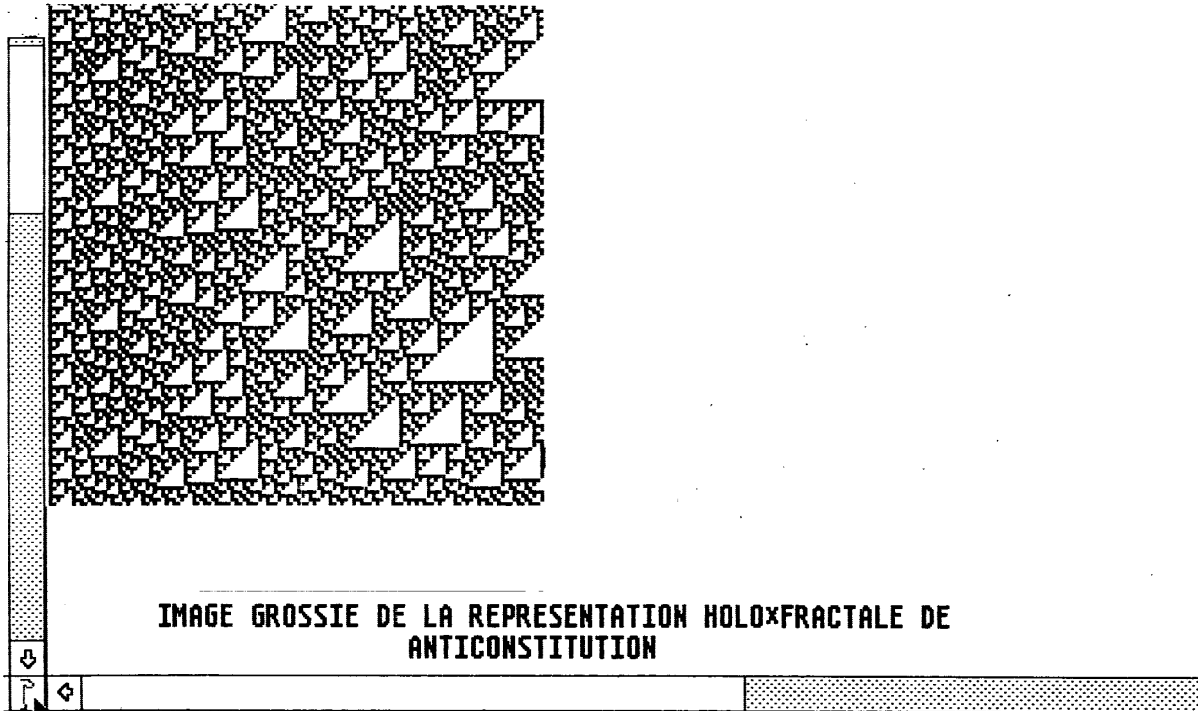
Déclaration préliminaire de Gérard: «*L'idiome \ existe en APL depuis plus de 15 ans... J'ai surtout découvert son importance primordiale unificatrice. Evariste Galois, il y a plus de 160 ans, a travaillé sur les sommes modulo 2. Ce que j'ai fait depuis plus de 10 ans a consisté:*

- 1) à comprimer tous les algorithmes possibles en cherchant "leurs noyaux incompressibles",*
- 2) puis soupçonné et fort surpris, dès 1989, que TOUS les algorithmes pourraient bien avoir ce même noyau, une fois franchi le "mur du son" du binaire,*
- 3) vu et étudié le rapport entre cette propagation et la physique la plus fondamentale,*
- 4) recherché dans les travaux des autres les compatibilités et incompatibilités (il n'y en a aucune, sauf avec les nombres et continu),*
- 5) recréé la mathématique nécessaire durant ces 4 dernières années, et inventorié de nombreux domaines d'application dont certains vont maintenant être décortiqués à fond».*

C'est ce que j'avais de mon côté pressenti depuis longtemps et j'avais plaidé dès 1980 pour une nouvelle branche à créer des mathématiques, suite à l'abandon de l'axiome du continu qui n'a aucun sens réel en physique: Ainsi furent proposées, mais non développées "les mathématiques repagulaires", le qualificatif venant du mot latin *repagula* qui veut dire saut. C'est Gérard Langlet qui a ensuite repris le flambeau avec le succès que l'on sait.

Structure du pariton

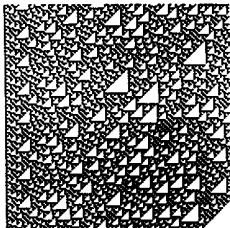
C'est une matrice creuse en tapis de Sierpinski. fig. 2.



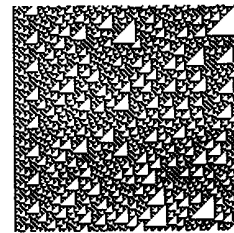
Symétrie du pariton

0 1 est identique à 1 0, mais après propagation, l'inverse n'est pas le symétrique du direct. fig. 3.

16 caractères (cadrage à gauche) ? Marcel Locquin



<- direct PARITONS inverse ->



```
Données (ou dernière ligne du Pariton ):
01001101011000010111001001100011011001010110110000100000010011000110111101100011
01110001011101010110100101101110000000000000000
Héliçon:
01011110001110010010101000101001001111000011011101111011010010110011000000110011
001100100010011000111010001110100000101101000011
Cogniton:
10101001000100101110011100010010110010100010111000100100100001000001000100011000
1100110010100011100110011000111000000000000000000
Vérification de la réversibilité de l'Héliçon:
Données (ou dernière ligne du Pariton inverse):
01011110001110010010101000101001001111000011011101111011010010110011000000110011
001100100010011000111010001110100000101101000011
Héliçon:
01001101011000010111001001100011011001010110110000100000010011000110111101100011
0111000101110101011010010110111000000000000000000
```

Intégrale de parité

Elle permet une exploration chronomodale et spatiomodale du pariton. Les événements sont déterminés dans leurs déploiements dans l'espace-temps par les conditions initiales, autrement dit le "cogniton", résultat de l'intégrale de la parité à partir de la séquence initiale, est déterminé entièrement par ce pariton de départ.

«Arrêt de Turing : On peut "rétrodirer" le pariton à partir de tout parcours de type saut de puce traversant le pariton de gauche à droite; un seul bit par colonne est suffisant».

«Cogniton: Tout cogniton C de S est aussi une séquence dont le cogniton est S. C et S sont des inverses spatio-temporels (mais C et H aussi, et H et S également)».

« Intégrale de Parité, Pariton: Le NIL est la colonne de gauche d'un pariton; il contient SOIT toujours 0 si la première parité est 0, SOIT toujours 1 si la première parité est 1. le Zébron suit toujours une colonne de 1. Pour éviter cette difficulté, je considère maintenant des séquences signifiantes, dont le premier bit est 1; alors le NIL contient toujours 1. Si une séquence débute par 0, son anti-séquence est toujours signifiante».

Parité et radioactivité

La théorie de la radioactivité élaborée dans les années 1930, postulait une invariance dans les interactions C.P.T., prises séparément: conjugaison positon-négaton, neutrino-antineutrino, réflexion d'espace et renversement de la flèche du temps. Cette position a été rendue intenable par le modèle de Lee et Yang conforté par l'expérience de Mme Yang en 1957. Il n'y a pas dans la réflexion relativiste conservation de la parité.

Une acrobatie mentale a conduit les physiciens de cette époque à admettre que c'était la conjugaison de charge qui était en défaut alors que subsistait l'invariance parité-temps. Toutes autres ont été les conséquences de l'hypothèse de Gérard Langlet dans sa théorie de propagation asymétrique de la parité.

Les fractals

Un fractal possède, dit-on, par définition, une homothétie interne, c'est à dire que le même motif formel se répète à des échelles emboîtées différentes.

Un fractal simple, celui de Witten et Sander (Phys. rev. letters 47, 400 1981) va nous permettre de mieux en comprendre l'essentiel. En version bidimensionnelle sur un réseau carré on le construit en convenant qu'une particule lâchée au hasard à l'intérieur d'un cercle, se propage, en suivant les lignes ou les colonnes, que sur les noeuds, adjacents ou non, d'un réseau carré de pas 1, ce qui est facile à construire sur ordinateur. A chaque pas on tire le pas suivant de façon aléatoire et on arrête le tirage lorsque la propagation touche le cercle. On obtient ainsi un amas fractal qui est une simulation simplifiée d'un mouvement brownien cumulé. Les agrégats colloïdaux d'or par exemple se construisent de cette façon, mais sans respecter la propagation rectangulaire. Ce fractal propage la différence de position uniquement. C'est ce type de fractal qui se construit lorsqu'on effectue une cristallisation en lame mince ou lorsqu'une décharge électrostatique ou claquage, se produit sur une lame non conductrice de l'électricité. C'est aussi ce type de fractal qui se produit dans l'effet de Lichtenberg ou dans l'effet Kirlian. On l'obtient également lors des dépôts cathodiques sur lames. La croissance d'une colonie bactérienne la reproduit également, ainsi que la migration des cellules d'un embryon sur le vitellus. Les fractals dentritique de digitation visqueuse ou de la fissuration gélatineuse en sont deux autres exemples.

Remarque de Gérard: *«Attention au mot "homothétie" vrai seulement pour les fractals réguliers (Génitons). Le terme correct est "auto-affinité" pour les paritons». Gérard ajoute: «Attention également à la notion d'"énergie potentielle", résultant de définitions macroscopiques qui ont conduit à des postulats en cascade. Le mot phase (transformé en "pseudo-phase") peut encore avoir un sens. Le mot "amplitude" est encore lié à la macroscopie. De même, "onde" doit devenir "pseudo-onde", ou, mieux encore "pseudo-ondelette"».*

Je ne suis pas Gérard sur ce point, car je ne connais aucun exemple expérimental en physique comme en chimie notamment, où les propriétés d'un niveau microscopique ne se retrouvent pas, au niveau macroscopique. C'est le changement du mode de manifestation de ces propriétés dû à la complexité qui en cache parfois l'auto-affinité.

Le départ prématuré de Gérard ne nous a pas permis de continuer notre confrontation sur ce point de divergence qui, pour moi est infirmé par l'expérience fondamentale de Zernike effectuée en 1934 et refaite depuis des milliers de fois avec de nombreuses variantes, sans qu'aucune en ait infirmé les conséquences,

tant microscopiques que macroscopiques. La reconstruction holographique macroscopique n'existerait pas s'il n'en était pas ainsi.

Applications biologiques de l'intégrale de parité

La fonction graphique SHAPES et ses dérivées AGGT et AGGC, méritent une attention toute particulière car elles sont des applications morphogénétiques de la propagation asymétrique de la parité.

On frappe au clavier la commande SHAPES3 *a b c*, dans laquelle *a b* et *c* sont des nombres dont *a* fixe dans l'image la symétrie, *b* la récursion et *d* l'intégrale. Le programme construit point par point le graphisme. Chaque figure en résultant est construite à partir de milliers, voire de dizaines de milliers de points. Les résultats sont étonnants. Les images obtenues sont des images de formes naturelles, simples ou complexes, techniques ou biologiques.

Par exemple, la commande SHAPES3 31 5 71 dessine une Diatomée centrique, SHAPES3 30 5 21 un poisson, SHAPES3 32 5 71 ou SHAPES3 5 5 82 dessinent deux autres poissons.

Perfectionnant le programme SHAPES est devenu AGGT, acronyme d'Action Génétique, Générale Trine, puis AGGC, acronyme d'Action Génétique, Générale, Conformationnelle.

La commande AGGT 0 dessine un triangle, La commande AGGC R S I dans laquelle R S et I sont des nombres entiers, permet de comprendre la genèse et de dessiner la plupart des formes biologiques et physiques naturelles.

C'est ainsi que AGGC 3 7 4 dessine en 24 576 points une figure géométrique ultra simple continuée d'un triangle équilatéral aux trois sommets duquel est attaché un petit triangle lui aussi équilatéral, AGGC 5 4 12 dessine un amas d'étoiles, AGGC 12 4 5 des molécules, AGGC 9 3 8 une rosace, AGGC 9 5 10 une rosace de cristaux, AGGC 27 3 31 un diaphragme iris, AGGC 17 2 5 une mosaïque, AGGC 31 5 71 une Diatomée centrique, AGGC 7 3 7 un flocon de von Koch, autrement dit un fractal en flocon de neige, AGGC 5 7 11 un champignon de l'ordre des Gymnascales, AGGC 47 5 6 une autre Gymnascale, AGGC 13 5 5 un détail de Gymnascale, AGGC 12 5 31 un Chitinomycète, champignon fossile, AGGC 8 5 47 une rafle de conidies, appareil reproducteur de Champignons asexués, AGGC 12 5 4 un Acritarce, qui est un micro-fossile, AGGC 30 5 21 un poisson, AGGC 32 5 71 un autre poisson, AGGC 5 5 82 encore un poisson.

Réflexions finales

Gérard s'est penché sur les raisons profondes qui ont retardé la découverte de l'intégrale de parité.

«Les opérateurs hamiltoniens (ou laplaciens) sont les grands coupables: ils ne décrivent correctement que des systèmes à deux corps (seul cas pour lequel le temps est, effectivement a priori réversible à cause de la "mise au carré", ce que l'on retrouve dans toutes les équations intégrées à partir de ces hamiltoniens, et pour cause). Les physiciens sont victimes d'une illusion due à l'auto-affinité du système: deux 1 non différents à un certain niveau d'échelle conduisent à voir à cette échelle un système à deux corps, alors qu'à l'échelle inférieure, les 1 sont différenciés et le sous-système est à N corps. Seuls les génitons sierpinskiens sont vraiment homothétiques. La matrice ci-dessous n'est pas un géniton, et pourtant, elle peut être prise de loin pour un géniton:

1 1 1 0 est perçu de loin comme: 1 0 1 1

1 1 1 1 0 1 comme: 1 0 1 0 0 1

Chaque carré 2 par 2 est perçu comme 1 si et seulement si ce carré contient "trois" n 1; sinon, il est perçu comme 0, car statique.

C'est pour cela que :

X X peut coder un système à deux corps au niveau supérieur, X Y alors que les deux X peuvent être très différents au niveau inférieur. Le Y au niveau supérieur est un 0 pour X, mais n'est pas 0 au niveau inférieur, son "lui-même".

Cette imbrication des relations pseudo-homothétiques va se retrouver partout (au niveau des échanges membranaires comme dans les transitions électroniques, les empilements cristallins et les étoiles doubles, et est la clé de la perception comme celle de la mémorisation: "la nuit, tous les minouchets sont gris").

C'est l'importance relative à chaque niveau qui compte (et qui code) et non pas l'amplitude. (Subtil, non? car cela risque d'expliquer aussi beaucoup de relations en psychologie et en sociologie, comme en tectonique des plaques, en traitement d'image, en linguistique et en musique)».

Alors que je lui faisais remarquer que la physique quantique avait conceptualisé la complexité du quanton en dualité onde-particule autrement dit phason-amplitudon en le formulant par une quantité complexe, il me dit:

«Attention au "pariton complexe". Le pariton non complexe se suffit à lui-même, car l'envers est déjà l'imaginaire de l'endroit et réciproquement, comme une image dans un miroir. Voir les propriétés de tous les génitons, opérateurs de rotation ternaires dans le plan complexe bien qu'ils ne contiennent rien d'autre que 0 et 1. Tous les génitons (et leurs inverses matriciels qui sont aussi leur carré et leur symétrique par rapport à la 2e diagonale) ont la propriété de "2j"5 racine cubique complexe de l'unité (les inverses ayant la propriété de "2j"2, conjugué et carré de "2j"5, ce dernier étant aussi le conjugué et le carré de "2j"5). "2j"5 est beaucoup plus important que "2i"5.

Attention aussi à l'"espace relativiste" qui risque fort d'y laisser des plumes (il est trop tôt pour oser critiquer...). Le réel perçu a "sdeux"n inverses, comme un signal asymétrique a, par transformée de Fourier, une partie réelle et une partie imaginaire. Cette "bi-transformation" joue probablement en biologie un rôle considérable».

C'est ainsi que Gérard a pu exposer le processus encore hypothétique qui permet aux cellules rétiniennes d'effectuer la transformée de Fourier du signal reçu pour pouvoir le transmettre, après compression, sans perte d'informations, par la voie du nerf optique dont le débit est bien trop faible pour pouvoir transmettre en temps réel les millions de pixels captés par la rétine dans une succession d'images.

Conclusion

L'oeuvre de Gérard Langlet ouvre la voie à de nouveaux champs immenses de recherche, non seulement dans le domaine de la programmation informatique, mais aussi et surtout dans les applications de l'intégrale de parité à de multiples domaines.

En collaboration avec de nombreux collègues, nous l'avons personnellement appliquée en biologie, principalement lors de la simulation de la morphogénèse génétique, dans l'analyse des perceptions et manifestations psychosensorielles avec toute l'équipe de recherche de l'AIR-ALCE dirigée par Frédérique Hourtoulle-Rollet, dans l'étude de la simulation informatique des processus de mémorisation et ressouvenance avec Fabrice Papy, de la simulation de la filtration des fréquences spatiales par percolation sierpinskienne.

Après le bon millier d'isomorphismes morphogénétiques techniques et biologiques que j'ai construits à l'aide de la fonction AGGC R S I, des millions sont encore à découvrir, ce qui nous donnera de nouvelles clés pour analyser les hiérarchies enchevêtrés du code génétique. J'arrête ici cette liste pour me limiter aux domaines effectivement en cours d'exploration entre 1992 et 1997.

Voie bibliographique

Une revue générale des travaux de Gérard Langlet a été publiée par A. Delmotte dans le numéro 23 de Juin 1997 des Nouvelles d'APL. On pourra y ajouter: Un S.O.S. à Formica, texte inédit de Gérard Langlet, paru dans le N° 24 de notre Revue.

Il faut aussi consulter FACTOTUM-MYCOLOC, progiciel taxinomique et linguistique polyvalent de Marcel Locquin et Gérard Langlet, UITF 1991, Rueil-Malmaison et Jouy en Josas. Dans les 51 pages de cette publication on trouvera une liste complète des commandes des fonctionnalités essentiellement taxinomiques qui y sont contenues, arrêtée à la date de Décembre 1990 .

Je rappelle les domaines explorés: Documentation. Programmes et accessoires indispensables. Commandes générales. Polices non latines. Editeur de minigraphes. Impressions et copies d'écrans. Chargement. Sauvegardes. Travail taxinomique: index et indices. Créations de textes. Graphismes. Mesures et précisions. Objets et tableaux. Médical. Mathématiques. Arbres, hiérarchies. Itinéraires. Molécules. Messages d'erreurs. Jeux scientifiques, simulations. Modes de calcul des indices taxinomiques. Conjugaisons, translittérations, déclinaisons.

Programmes annexes. A la fin on y trouve l'analyse de l'essentiel des 63 documentations rédigées par Gérard Langlet. La Doc 63 est consacrée aux Symétries appelées par la commande WYCK en hommage à Ralph W. G. Wyckoff avec qui nous avons travaillé tous les deux, Gérard en cristallographie et moi-même en microscopie électronique.

Rueil--Malmaison, Novembre 1997



Chers APListes,

Bonne et heureuse Nouvelle Année 1998

La Rédaction

Ce siècle avait 99 ans...

par Bernard Mailhol

1 Présentation

Le problème de l'an 2000 est évident, voire simpliste : les systèmes informatiques - que nous avons développés - stockent la valeur de l'année sur 2 caractères, en oubliant de citer le siècle.

Mais lorsque ce siècle change ...

Ce texte reprend une présentation faite à la journée de l'AFAPL du 6 novembre dernier. Il a été créé à partir de cette présentation, des commentaires ayant été ajoutés entre chaque texte présenté.

Le support de cette présentation est aussi disponible en <http://www.mailhol.com/pub/afapl>, sous forme de pages html.

Sur ce site, vous pourrez aussi trouver l'ensemble des fonctions citées dans cet article.

2 Bonne année

Tout commence au 1er janvier 2000 !

2.1 Le problème

- Habitue "sociale"

Nous avons tous l'habitude, quand nous écrivons et datons, d'écrire les dates en oubliant le siècle. Qui irait dater du "6 nov 1997" ? L'habitude fait que nous datons plutôt "6 nov 97".

La forme complète (1997) a même une connotation quelque peu maniérée et, jusqu'à ce jour, d'emploi peu agréable.

Il est évident que les habitudes de programmation reprennent ces habitudes de vie sociale. Les saisies de dates, de la même façon, ne peuvent imposer aux utilisateurs de saisir 4 chiffres.

A partir de ce moment, ce n'est pas au programme d'inventer le siècle.

- Économie de place en mémoire (dans les temps anciens)

D'autre part, les applications que les entreprises utilisent ont parfois plus de 10 ans d'âge, voire 15 ans. Le développement d'une nouvelle application demande un tel effort de la part de ses concepteurs fonctionnels qu'une entreprise modifie une application plutôt que la renouvelle.

Il y a quelques années, les coûts des matériels étaient très supérieurs à ce qu'ils sont maintenant, les technologies de stockage beaucoup moins élaborées. La capacité d'une bande magnétique des années 80 (on ne dit pas 1980 !) est très inférieure à elle des unités de sauvegardes (même si on se limite aux unités ayant la même fiabilité que ces bandes magnétiques).

Le coût de la mémoire, aussi était faramineux. Il a chuté dans de grandes proportions.

Vis à vis de ces problèmes, l'économie de stockage était très recherchée; les systèmes stockant une date sous forme de caractères se limitaient évidemment en supprimant le siècle de ce stockage.

Limiter l'année à 2 chiffres

Ces bonnes raisons ont fait que, pendant de nombreuses années, les humains - et leurs systèmes informatiques - ont raisonné en oubliant que le siècle pouvait changer.

Le temps étant ce qu'il est, il progresse ... et ne peut suspendre son vol. Le siècle va bientôt se terminer.

On passe ainsi de 99 à 00

2.2 Risques

1 - Le risque - **essentiel** - repose alors dans la comparaison des dates. La valeur 0 étant inférieure à la valeur 99, les comparaisons habituelles indiquent que la date 020100 est antérieure au 291299.

Quand le lendemain devient le passé, nos calculateurs s'affolent

Tous les systèmes qui calculent des durées vont ainsi brutalement se trouver en face de durées négatives, ou - selon la programmation et la prise en compte de valeurs négatives dans la représentation de leurs données - se trouver en face de très grandes valeurs. Aucun algorithme ne peut résister à ces conditions.

2 - Il est aussi possible que l'ajout de 1 à la valeur 99 provoque un dépassement de capacité lors du stockage de la réponse. Cet arrêt brutal du calculateur a alors un grand intérêt : il ne provoque pas de faux résultats !

Dans ce cas, l'alerte est immédiate.

3 - Valeurs spéciales

Un autre problème vient aussi de la signification particulière donnée à certaines dates, dans les habitudes de programmation.

a - La valeur 00 est un joker indiquant que l'année est inconnue. Ce sera alors le cas de l'an 2000.

Une valeur 0000 est alors légale, car l'an 0 sort effectivement de l'attribution de nos calculateurs !

b - La valeur 99 peut être utilisée pour indiquer que l'entité décrite n'a pas de fin.

La valeur 9999 est alors légale, car l'an 9999 sort effectivement de l'attribution de nos calculateurs.

2.3 D'où vient le problème ?

1 - Ce n'est pas souvent un problème de matériel (dans le cas des machines programmables)

Les calculateurs dont nous disposons aujourd'hui permettent tous le passage à l'an 2000. Il se peut qu'ils s'arrêtent un peu plus tard, mais la limite des horloges est un autre problème, à traiter en son temps.

2 - Problèmes dans des ordinateurs "embarqués".

Ces calculateurs embarqués sont assimilés à du matériel : nous n'avons aucun pouvoir sur leur fonctionnement. Mais ils peuvent comprendre de la programmation, alors *microcodée*. Il s'agit bien de programmation, le passage à l'an 2000 doit être examiné avec soin.

Supposons qu'un matériel effectue des tests de durée, par différence entre deux dates - on ne peut faire autrement. Ce logiciel, au passage à l'an 2000, peut alors trouver une durée aberrante.

Que va proposer la commande de l'ascenseur quand elle se rendra compte qu'il a fallu (moins 1) siècle pour parcourir un étage !

Ne prenez pas l'ascenseur le soir du 31 décembre 1999

3 - Ce n'est pas souvent un problème de système d'exploitation

Ce problème est de fait assez mineur : tous les systèmes d'exploitations maintenus sont ou seront bientôt capables de franchir le siècle.

Les problèmes concernent les systèmes non maintenus, qui ne peuvent rien garantir, et risquent de ne plus fonctionner **du tout** dès le 1 janvier 2000. Les applications qu'ils supportent doivent impérativement être migrées.

4 - Problèmes dans les outils généraux

Cette famille de problèmes est assez répandue, et nécessite une remise en question des outils utilisés.

Ces outils doivent être mis à niveau, *en même temps* que vos applications, ou bien être dupliqués sous un autre nom.

Prenons par exemple un outil donnant une liste de fichiers, avec leur date de création ou de dernière mise à jour. Vous appelez cet outil pour extraire la date en question. Si la présentation de cette date change, la compatibilité ascendante de l'outil n'est pas assurée, vos anciennes applications sont en erreur.

5 - Problèmes dans vos applications

Ces problèmes sont réels, et la mise à niveau de ces applications ne peut suivre de méthode générale.

Il faut reprendre chacune de ces applications, les analyser, les modifier les tester (*Attention, danger*).

Ce travail est directement lié au nombre et à la taille des applications à analyser.

6 - Problèmes dans les bases de données

Les bases de données contiennent les données, entre deux exécutions de vos applications. Si vos applications ignorent le siècle, il y a des chances (des risques) que vos bases de données souffrent des mêmes lacunes.

Evidemment, les systèmes modernes indiquent le siècle, mais quelles sont les règles utilisées pour "ajouter le siècle" aux dates n'en disposant pas ?

Une modification du format de ces bases implique un lourd travail, la méthode utilisée doit non seulement tenir compte de la migration de la base active, mais aussi des bases historiques, décrivant des 10, 15 ou 20 dernières années.

2.4 Quelles conséquences

Quel est l'impact de ces problèmes sur le fonctionnement d'une entreprise ? Donnons-en seulement deux exemples :

- Vous ne pourrez plus connaître les factures émises depuis trois mois, et non encore payées (elles ne sont pas encore émises)

- Vous ne pourrez plus appliquer un tarif (en 2000, l'année 00 est l'année interdite)

Votre gestion s'arrête

Il se peut aussi que vous utilisiez de la programmation embarquée. Dans ce cas, vous risquez l'arrêt complet du matériel.

Votre production s'arrête

2.5 Attention aux tests

Vous avez commencé l'analyse de vos applications, et commencez une migration. Faites attention, les tests peuvent entraîner des conséquences imprévues !

- On ne se projette pas impunément dans le futur.

Pour effectuer ces tests, vous devez changer la date de votre machine, pour vous trouver en l'an 2000, ou après. Tout d'abord, le fait d'être en l'an 2000 peut rendre invalides les fichiers que vous utilisez, si ceux-ci sont protégés par une durée de rétention.

Si, de plus, votre système est complet et sécurisé, il peut effacer ces fichiers caduques avant de vous laisser passer la première commande.

De toutes façons, il vous aura fallu vous connecter, et il y a des chances pour que vos mots de passe soient eux aussi à validité limitée, et leur expiration vous permettra de contempler les messages vous annonçant l'effacement de tous vos fichiers, sans pouvoir agir.

- Si vous avez pu vous projeter au siècle prochain, il vous faut pouvoir en revenir.

Voici deux exemples :

*Une chaîne industrielle a été testée pour le passage en l'an 2000. Elle s'est arrêtée sans surprise, au passage à l'an 2000.

La surprise est venue quand on n'a pas pu la redémarrer une fois revenue en 1997.

*Après un essai, un automate de lancement de travaux peut attendre ... le lendemain du dernier passage !!

Plus généralement, si une application a mis à jour un fichier, son usage futur dans le passé devient alors aléatoire *Mais qui a tué Harry ?*

Plus généralement, les tests de passage en l'an 2000 donnent aussi l'occasion de tester la reprise de la machine sur crash complet. (Toutefois, ce dernier exercice fait partie des procédures associées aux sauvegardes).

3 Le projet de migration

Il est agréable de faire un peu d'humour vis à vis de ce cataclysme. Cet humour peut en fait être un moyen de défense contre une grande angoisse.

Le mieux est quand même de réagir, et de migrer.

3.1 Qui est concerné ?

Toutes les entreprises.

3.2 Comment faire ?

Aucune méthode générale

On ne peut trouver une méthode unique pour traiter ce problème. Des outils peuvent participer à son analyse (par exemple, recherche dynamique des interrogations de l'horloge du système). La solution est au cas par cas.

• Dans tous les cas, il faut reprendre tous les programmes un par un.

La reprise d'un programme pose deux questions importantes :

a - *Dispose-t-on d'une documentation correcte, et testée ?*

L'intérêt de cette documentation est immense. Le moment de la migration n'est peut-être pas le meilleur pour essayer de comprendre l'application.

b - *Dispose-t-on du source, s'il est compilé ?*

Cette contrainte est évidente, mais les systèmes de suivi de configuration logicielle, s'ils sont peu utilisés aujourd'hui, étaient inconnus il y a quelques années.

Des programmes très anciens, mais donnant toute satisfaction, et ne nécessitant aucune maintenance peuvent être victimes de cette qualité, car plus personne ne sait où trouver les sources, dans leur dernière version...

Un autre cas de sources indisponibles est celui de l'usage de produits logiciels non maintenus, mais pouvant quand même fonctionner. Il faut alors reprendre complètement l'application, ou négocier avec le constructeur une

évolution en-dehors de ses plans de maintenance, en-dehors de ses engagements.

- Revoir la topologie des bases de données (quid des archives de 10 ans ?)

Si une date est stockée sans le siècle, vous devrez introduire cette notion, et modifier la définition de la base.

Il faut alors reconfigurer la base, établir un plan de migration de l'application actuelle, établir un plan de reprise des archives, comme l'impose la loi française pour certains types de données.

- Si un logiciel est acheté, qui a la charge de l'évolution ?

Nous avons déjà évoqué ce problème concernant les produits dont le constructeur a délibérément abandonné la maintenance, en prenant toutes les précautions nécessaires, mais dont vous n'avez pas (encore) tenu compte car vous utilisez encore le produit.

En ce qui concerne les autres produits, nous ne savons pas aujourd'hui qui a la charge de la migration de l'application. Certaines rumeurs évoquent les logiciels achetés après 1990, mais il ne s'agit en aucun cas d'une garantie décennale, car le logiciel ne s'use pas, les conditions du passage du siècle étaient connues, aussi bien en 1980 qu'en 1997...

Il s'agit de négocier avec votre fournisseur. Si il est amené à déposer son bilan, sachez que vos applications ne pourront **plus jamais** être migrées. Un peu de raison s'impose.

3.3 Quand commencer ?

Ces travaux de migration sont très longs, ils demandent une bonne organisation, et beaucoup de travail. Ils peuvent imposer la reprise complète de certaines applications, ce qui demande plusieurs années pour fournir une nouvelle application effectivement utilisable.

En caricaturant un peu, on peut indiquer qu'il faut commencer **dès 1995**.

Il est donc trop tard si vous n'avez pas encore commencé.

Pour le moins, ces travaux ne se dérouleront pas avec la sérénité nécessaire. Leur issue peut être incertaine, si des applications se révèlent rebelles à toute migration.

3.4 Spécificité de ce projet

Ce projet informatique présente des spécificités :

- Il coûte cher
- Au mieux, il ne rapporte rien
- Il ne souffre aucun délai
- Il demande beaucoup de travail
- Peu de personnes sont disponibles

Examinons chacune de ces propositions :

- Il coûte cher

En effet, ce projet coûte très cher. Même si peu de travaux sont effectivement nécessaires, il aura fallu au moins s'assurer de la capacité de fonctionnement au siècle prochain, et les tests ne sont pas évidents.

Si des travaux sont effectivement nécessaires, leur coût peut être important.

Si une application doit être reprise, ces coûts sont énormes.

- Au mieux, il ne rapporte rien

Le but de cette migration est de permettre aux applications informatiques, aux chaînes de production de pouvoir fonctionner ... comme avant.

Aucune valeur ajoutée n'est donnée. Il s'agit en fait de permettre la poursuite de l'activité de l'entreprise !

- Il ne souffre aucun délai

La plupart des projets informatiques ne sont pas lancés dans les délais annoncés, et les Directeurs Informatiques savent très bien gérer ces retards.

Pour des raisons évidentes, la migration "An 2000" doit être prête dès le 1er janvier.

Un changement de Directeur ne permet pas de faire avancer le problème.

- Il demande beaucoup de travail

Ces travaux exceptionnels demandent des compétences nouvelles. Le travail à effectuer est proportionnel à la durée de fonctionnement du centre.

Ces travaux supplémentaires sont très volumineux.

- Peu de personnes sont disponibles

Enfin, tous les projets "An 2000" ont la même échéance !

Toutes les équipes sont mobilisées exceptionnellement. Les rumeurs évoquent le rappel de nombreux retraités.

3.5 Nouveaux Risques

Après les premières migrations, des risques nouveaux ont été rencontrés.

- Certains fournisseurs peuvent donner de fausses informations.

La migration commence par une phase préliminaire d'enquête sur les capacités des produits à franchir l'an 2000.

Il s'est trouvé qu'un fournisseur fasse une réponse masquant le problème. Ainsi, à l'une de ces questions, un fournisseur a répondu : *"Nos dates sont bien sur 8 positions"*, en oubliant de préciser que deux "/" sont compris dans ces 8 caractères.

De telles réponses peuvent produire un retard dans un projet ordinaire. Dans le projet "An 2000", elles produisent aussi un retard, mais ce retard peut être fatal.

- Certaines transformations peuvent être effectuées par des sous-traitants "achetés".

La migration des programmes, un par un, demande beaucoup de main d'oeuvre. La sous-traitance est alors souvent rencontrée. Mais attention, il faut contrôler le travail effectué, ce qui demande encore du temps.

La rumeur évoque un programme, "*migré an 2000*", dans lequel le sous-traitant avait introduit une bombe logicielle.

Que cette rumeur soit vraie ou fausse, elle est plausible, vous devez vous protéger.

•Rechercher vraiment partout.

Les blocages de l'an 2000 peuvent intervenir dans tous les systèmes, que l'informatique soit apparente ou non, qu'ils soient d'usage fondamental, ou d'usage secondaire.

Leur défection viendra de toutes façons verrouiller votre production ou l'application de vos procédures.

4 Les applications en APL

Nous avons évoqué les problèmes généraux posés par le passage à l'an 2000.

En ce qui nous concerne, notre casquette d'APListe nous conduit à faire migrer des applications écrites en APL.

Nous nous rendons compte, une fois de plus, que l'APL fait de nous des privilégiés.

4.1 APL et l'an 2000

Tout d'abord, APL propose **une seule primitive** pour traiter des dates. De toutes façons - avec si nécessaire la mise à niveau appropriée du système et de l'APL - cette primitive permet de traiter l'an 2000.

•TS

1997 11 6 15 35 200

Le temps est donné, de base, en *année mois jour minutes secondes millisecondes*. L'année est systématiquement donnée correctement (par un nombre supérieur à 1900)

Cette primitive a été définie, depuis le début voici une trentaine d'années, pour rendre l'année, correctement

L'APListe, influencé par son APL devrait alors utiliser un codage de l'année sur 4 positions.. Il peut bien sûr avoir outrepassé cette règle, mais en tronquant délibérément l'information rendue par le système.

Il doit alors migrer ses applications, et commencer par se procurer la documentation et les sources du système.

L'APL étant un langage interprété (avec un degré de pré-compilation variable selon les systèmes), tout développeur dispose *effectivement* des sources des applications, à jour.

La première étape est de rechercher les usages du temps. Il suffit alors de rechercher les usages du •TS. Cette recherche est unique.

Faites attention, car vous utilisez souvent des outils ou procédures externes à l'APL en lui-même, et ces outils peuvent être dépendants de la date et son expression. Encore une fois, il y a des risques cachés.

4.2 Le monde de l'APL

La nature du langage APL nous conduit à privilégier certaines catégories d'applications.

On effectue souvent des travaux de simulation, de projection sur les prochaines années.

Nous avons de fait été confrontés au problème de l'an 2000 depuis plusieurs années. Depuis longtemps, nous avons (presque tous) mis en place des systèmes de prévisions à 15 ans. Depuis 15 ans, nous avons été confrontés au problème de l'an 2000. Depuis 15 ans, nous y prenons garde.

Nous avons souvent déjà traité l'An 2000

De plus, la structure d'une programmation APL fait que toutes les dates sont souvent traitées dans peu de fonctions, le stockage (numérique) acceptant aussi bien les années comprises entre 72 et 99 qu'entre 1972 et 2099

Ce stockage numérique fait que la migration, si elle ne se réduit pas à une revue de code, se fait souvent en compatibilité ascendante des bases de données.

4.3 Contraintes apportées aux dates

A l'occasion de la migration de vos applications vous pouvez être amenés à modifier la représentation interne des dates. Cette représentation interne doit satisfaire à plusieurs conditions

1 - Permettre une saisie

Vous devez disposer d'une fonction de codage d'une date issue du •TS en une valeur interne.

2 - Permettre une édition

Vous devez disposer d'une fonction de décodage, à partir d'une représentation interne. Attention, les éditions doivent pouvoir être effectuées l'année étant aussi bien représentée par 2 chiffres que par 4 chiffres.

Dans certains états, la taille de la ligne peut ne pas absorber les 2 caractères supplémentaires pour chacune des dates. Cet allongement de la ligne peut conduire au débordement d'une page, à l'usage d'une police vraiment trop petite
...

3 - Permettre une comparaison

Deux opérations sont importantes sur les dates. La première consiste à comparer deux dates entre elles, et ensuite de déterminer des permutations à fins de tri selon une séquence croissante ou décroissante.

4 - Permettre un calcul de délai.

L'autre opération est un calcul de délais. Ces délais peuvent être exprimés de diverses manières.

5 - Offrir une plage de dates suffisante, une précision suffisante.

Ce choix est important, et peut conditionner une application.

La plage de dates peut concerner quelques années ou dizaines d'années.

Un système traitant de personnes doit pouvoir représenter une date commise entre 1880 (éventuelle année de naissance) et 2080 (éventuelle année de retraite), avec une précision de la journée.

D'autres systèmes peuvent devoir stocker la date d'événements compris entre 990 et 2020, mais avec une précision de la seconde.

Toutes ces contraintes ne conduisent pas forcément à une représentation unique des dates dans une application.

4.4 Les méthodes de rattrapage

Plusieurs méthodes permettent d'inventer le siècle associé à une date, et de la stocker. Ici encore, aucune méthode générale n'est proposée.

1 - Changer la codification des dates, avec détection du codage précédent.

Cette méthode conserve la représentation initiale, et n'impose aucune transformation sur les données externes à l'application.

2 - Fenêtrage (glissant) sur un siècle (entre année - 20 et année + 80 par exemple)

Ce fenêtrage est possible sur certaines applications, il complique les opérations de comparaison et de tri, les calculs de délais.

3 - Changement du format externe (si codage en caractères par exemple) *(le plus lourd, car on peut difficilement utiliser les bases précédentes)*

Certains fichiers externes contiennent des données sous forme de caractères, pour compatibilité avec des systèmes non APL par exemple. Dans ce cas, les problèmes sont beaucoup plus aigus, car ils mettent en cause la topologie des fichiers.

4.5 Saisie d'une date

L'un des problèmes que nous avons à traiter est celui de la saisie d'une date. Cette saisie doit éliminer toute ambiguïté, et doit permettre une saisie de toute date, conformément aux habitudes de chacun.

Pour ma part, je dispose d'une fonction permettant une saisie sous différentes formes :

1 - Trois formes principales sont reconnues :

a - Trois valeurs séparées par des espaces (ex: 6 11 97)

b - Trois valeurs séparées par un caractère quelconque (ex : 6/11/97)

c - Trois nombres accolés (ex : 061197)

2 - L'expression de l'année sur 2 ou 4 positions.

L'année peut être sur deux positions pour les années 1980 à 1999. Elle est sur 4 positions dès l'an 2000

Par exemple, les deux saisies suivantes sont valides : 6/11/97 et 6/11/1997.

3 - L'ordre des informations à choisir par l'utilisateur

a - Présentation *à la française*

La date comprend le jour, le mois, l'année. Cette date peut être ambiguë si le contexte est international, et que certaines parties de l'application placent le mois en tête.

b - Présentation selon *ISO*

La date comprend l'année, le mois, le jour (c'est d'ailleurs ce que nous propose APL).

Cette forme ne permet aucune ambiguïté, car la présence de l'année en tête est immédiatement identifiable, et est alors suivie, quelles que soient les habitudes, par le mois et le jour.

Ainsi, toutes ces saisies sont valides :

04/07/2006	04/07/97	04/07/1997
04072006	040797	04071997
2006/07/04	97/07/04	1997/07/04
20060704	970704	19970704

... et ne présentent aucune ambiguïté.

Exemples de fonction de saisie

1 - Fonction `srvCDT` Contrôle de saisie

```
rz½srvCDT CH;N;I;M;•IO
à compiler une seule date (vecteur alpha)de 1980 ... 2059
à retour en rz
•IO½0
,0õ'Caractères invalides' srvERR~^/CHî'0123456789/-.:_'
((~CHî'0123456789')/CH)½' ' à ne laisser que les chiffres
,0õ'Date absente' srvERR 0=æCH½ebl CH
,0õ'trois valeurs' srvERR~(æN½,~CH)îl 3
à formes possibles, ... reconnaître
```

ä jj mm aa ou jj mm aaaa (aa possible jusqu'en 1999)
 ä aa mm jj ou aaaa mm jj(id)
 ä aammjj ou aaaammjj
 ä jjmmaa ou jjmmaaaa
 , (3=æN)/L0
 I½(í0 100 100~N),[0](í0 100 10000~N) ä deux cas possibles
 ä placer l'ann,e en tete
 (MöI)½è(M½I[;2]=©/I)öI
 N½,((I[;1]î1+î12)^(I[;0]<100)ëI[;0]ð1980)öI
 ,0ö'Date invalide' srvERR 3ôæN
 ä une date sous forme de trois nombres
 L0:I½(N[2]=©/N)ãN(èN) ä ann,e en tete
 ,0ö'Date invalide' srvERR(I<1),(I>2050 12 31)
 ,0ö'Ann,e invalide' srvERR~((ÆI)î80+î20)ë((ÆI)î1980+î80)
 ä I est une date aaaa mm jj
 ,0ö'Date incorrecte' srvERR ý1=cta I½ymd I
 rz½(î0)I

2 - Fonction ymdTransformer une date complète

Z½ymd V
 ä soit un vecteur de trois ,l,ments aa mm jj, rendre aaaa
 mm jj
 ,(1=ææV)/L0
 ,0 Z½(yr 1Æ[1]V),1Ç[1]V ä tableau
 L0: ,0 Z½(yrÆV),1ÇV ä vecteur

3 - Fonction yrTransformer une année

YYYY½yr YY;K
 YYYY½(æYY)æK+(K<100)ö1900+100ö(K½,YY)<50

L'ensemble (complet) se trouve sur le Web, en fichier *atf*

5 Meilleurs voeux 2000 !

Nous disposons encore de près de 800 jours (en Novembre 1997) pour traiter ce problème de migrations.

5.1 Surprise !!

Meilleurs voeux 1999

Car, dès le premier janvier 1999, nous sommes déjà dans *l'année de l'infini !!*, les applications vont commencer à perdre la tête.