Faculté des Sciences

INTRODUCTION A L'INFORMATIQUE

H.P. Garnir & F. Monjoie

Notes de cours

Année 2005-2006

Préface

Ce texte concerne le cours "Introduction à l'informatique" destiné au second baccalauréat en Biologie et Géologie et à la première licence en Biochimie.

Notons que les travaux pratiques se déroulent actuellement sur des micro-ordinateurs fonctionnant sous le système d'exploitation Windows de Microsoft. Les logiciels installés sur ces machines permettent de se familiariser avec le traitement de texte, les tableurs et différents logiciels de traitement de données. Les étudiants apprendront à rédiger un rapport présentant, sous forme attractive, des données numériques préalablement analysées. Toutes les machines sont reliées à l'Internet. Une part importante des travaux pratiques sera orientée vers la maîtrise des nouvelles possibilités offertes aux scientifiques par ce nouveau moyen de communication. Tous les étudiants sont amenés à utiliser le courrier électronique et le WWW. Nous indiquerons aussi comment trouver rapidement des informations pertinentes sur le Web.

Depuis 1999, les notes de cours ainsi que des documents relatifs aux exposés oraux se trouvent sur un serveur WWW (http://www.ulg.ac.be/ipne/info). Ce serveur est local et son accès est limité par un mot de passe. Le "user name" est "info" et le "password" est "ok".

Janvier 2006

Henri-Pierre Garnir
I.P.N.A.S.
Sart Tilman B-15
B-4000 Liège
Tél.: 4/ 3663764 & 3663690
hpgarnir@ulg.ac.be

Francine Monjoie
Mathématiques Générales
Institut de Mathématique B-37
B-4000 Liège
Tél. : 4/3669432
Francine.Monjoie@ulg.ac.be

C H A P I T R E I

Concepts fondamentaux

1.- INTRODUCTION

L'informatique est l'art, la technique ou la science qui consiste à manipuler des informations à l'aide d'un outil, l'ordinateur. L'informatique a pour objet de définir des algorithmes qui permettent de modifier la vision que l'on a d'un problème, ou d'extraire d'une grande quantité d'informations mal structurées, de nouvelles connaissances plus utiles.

Les outils de l'informatique sont les ordinateurs. Actuellement, on utilise presque exclusivement des ordinateurs digitaux (dans lesquels les informations sont codées sous forme d'états discrets) et qui fonctionnent selon les principes de la machine de Von Neumann (définie dans les années quarante). Cette machine comprend deux parties, une unité logique et arithmétique banalisée et un magasin ou mémoire qui contient des programmes et des données. Un programme décrit les opérations logiques à réaliser sur les données. Les deux idées neuves sont les suivantes.

• **L'unité logique est banalisée**. On ne sait pas lors de sa construction à quoi elle va servir, elle est seulement capable d'exécuter séquentiellement certaines instructions.

• **Les programmes et les données sont placés sur un pied d'égalité** dans la mémoire. Il est impossible de discerner les programmes des données si ce n'est par l'effet qu'ils ont sur l'unité logique.

Un ordinateur doit aussi comporter des organes d'entrée et de sortie qui permettent à l'utilisateur de placer ses informations dans la mémoire et de les y relire lorsque la machine les a manipulées.

L'informatique s'oriente depuis quelques années vers la micro-informatique individuelle. L'ordinateur individuel est utilisé par une seule personne à la fois qui décide seule de l'activité de la machine. Le prix des petits ordinateurs continue à baisser, ce qui accentue la tendance qui consiste à distribuer les machines plutôt que d'utiliser un gros système travaillant en temps partagé. Les ordinateurs sont mis à la disposition de chacun et connectés en réseau. Une connaissance suffisante de ces machines et des concepts qui en sous-tendent le fonctionnement contribue à démythifier le

phénomène de “l'informatisation de la société” et permet d’utiliser l’outil dans les meilleures conditions.

2.- LA PROGRAMMATION

Programmer consiste à construire un ensemble ordonné d'instructions qui, lorsqu'elles sont exécutées, produisent des effets précis et utiles sur les informations contenues dans un ordinateur. L'ensemble des instructions s'appelle un programme.

Préparer un programme destiné à un ordinateur est une tâche délicate et complexe. En effet, contrairement à l'être humain, l'ordinateur est incapable d'initiative ou de tolérance. Par conséquent, son programme doit être parfaitement explicite, doit couvrir tous les cas de figures possibles et doit entrer dans les moindres détails. Cependant, lorsque le programme s'exécute, l'ordinateur fait preuve d'une précision et d'une persévérance à toute épreuve.

Un autre truisme est de rappeler que l'ordinateur ne résout jamais un problème. Ce sont les hommes qui utilisent l'outil ordinateur pour aboutir à leurs fins. Sauf panne technique (extrêmement rare), toute "erreur" de l'ordinateur doit être comprise comme ayant une origine humaine.
Puisqu'un programme d'ordinateur doit être parfaitement précis, une grande part de la construction d'un programme consiste à analyser le problème que l'on désire résoudre, à le définir clairement, et à envisager les cas marginaux qui, bien que peu probables, peuvent perturber un schéma de solution. Il faut aussi considérer le jeu d'instructions dont on dispose afin de l'utiliser au mieux dans le contexte de la résolution du problème.

L'étape suivante consiste à coder le schéma de solution adopté dans un format que l'ordinateur peut assimiler. En général, le programme prend la forme d'un texte dont la syntaxe est extrêmement rigide pour se conformer aux directives d'un langage donné. Ce texte est tout d'abord écrit "dans" l'ordinateur grâce à un dispositif adéquat. A notre époque, on utilise généralement un clavier et un écran de moniteur (TV). Un programme préexistant, fourni par le constructeur de la machine et appelé Editeur de texte, aide le programmeur dans sa tâche. L'Editeur permet de faire "entrer" le texte du programme dans la mémoire et ensuite de le corriger très aisément.

Le texte du programme est ensuite traduit automatiquement par l'ordinateur en une suite d'instructions élémentaires qui lui sont spécifiques. Cet ensemble d'instructions peut être exécuté et il faudra vérifier que l'effet du programme sur les données correspond bien à l'attente du programmeur (phase du “debugging”, littéralement : "de la chasse à la petite bête"). Il sera aussi utile de composer un "mode d'emploi" du programme qui explique comment s'en servir et qui décrit son fonctionnement.

L'analyse d'un problème en vue de sa résolution par la voie de l'informatique implique l'adoption d'une nouvelle façon de penser. Les méthodes humaines traditionnelles, généralement basées sur l'intuition et le raisonnement (deux qualités qui font cruellement défaut à l'ordinateur), ne sont plus du tout applicables. Il faut acquérir de nouveaux réflexes qui tiennent compte des caractéristiques saillantes de l'ordinateur : habilité à réaliser des opérations répétitives simples avec précision et grande fiabilité dans ses facultés de mémorisation.

3.- QUE PEUT FAIRE UN ORDINATEUR ?

Un ordinateur peut traiter des informations. L'ordinateur est un outil qui nous aide à résoudre certains problèmes. Ces problèmes font généralement intervenir des symboles ou des signes qui nous sont familiers tels que les lettres de l'alphabet, les chiffres, les signes de ponctuation et quelques caractères spéciaux comme *,#,+... Nous pouvons introduire un ensemble de symboles dans l'ordinateur. Ce dernier nous rend un autre ensemble de symboles en relation avec ceux que nous avons introduits. Voici quelques exemples.

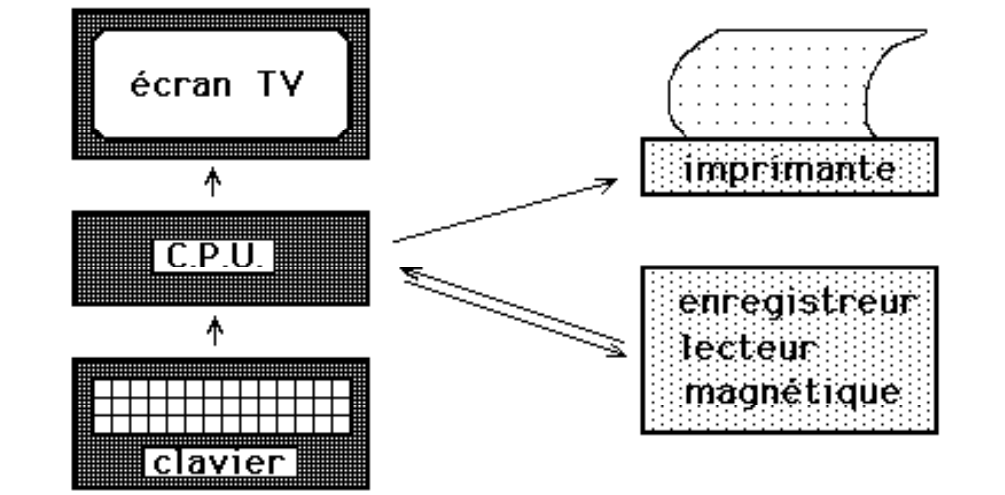
Caractères entrés	Caractères sortis
• Nombres représentant les dimensions d'une vitre.	• Prix de la vitre.
• Liste des livres empruntés dans une bibliothèque.	• Liste des livres ayant dépassé le temps de prêt.
• Le nom d'une personne.	• Son numéro de téléphone.
• Notation standard d'un déplacement sur un échiquier.	• Dessin de l'échiquier et du déplacement effectué.
• Nombres représentant une hauteur et une accélération.	• Dessin d'un atterrissage lunaire.

Nous n'examinerons pas en détails la façon dont l'ordinateur travaille mais nous apprendrons à lui donner les instructions requises pour qu'il exécute une tâche donnée. Il est cependant utile de connaître les différentes parties qui constituent un ordinateur.

4.- QU'EST-CE-QU'UN ORDINATEUR ?

Sous sa forme la plus schématique, un ordinateur se compose de trois parties ou unités.

Schéma d'un ordinateur



1) Une unité d'entrée (=input device) nous permet d'introduire des instructions et des données. Le clavier en est un exemple.
2) Une unité centrale (CPU = Central Processing Unit) traite ce qui a été introduit.
3) Une unité de sortie (=output device) nous transmet les résultats. Un écran ou une imprimante sont de telles unités.

Rien de très palpitant à première vue! Mais tout l'intérêt de l'ordinateur réside dans trois de ses caractéristiques, à savoir :
• sa capacité d'emmagasiner une grande quantité d'informations,
• sa rapidité de traitement,
• sa capacité d'assimiler un programme qui contrôle son propre fonctionnement.
Cette dernière caractéristique, de loin la plus importante, fera l'objet d'une grande partie de ce cours.

Un auxiliaire précieux : la mémoire de sauvegarde

L'ordinateur ne peut fonctionner que s'il est alimenté en courant. Dès que celui-ci est coupé, la plupart des ordinateurs perdent les informations qu'ils ont reçues. Il est donc intéressant de pouvoir sauver certaines informations utiles pour des tâches ultérieures de façon à les rendre à l'ordinateur autrement qu'en les tapant au clavier (ce qui s'avérerait vite fastidieux et risquerait d'introduire des erreurs...).
Cette mémoire de sauvegarde (=backing store) est généralement un support magnétique sur lequel on peut enregistrer des informations puis les relire grâce à un appareil enregistreur/lecteur ad hoc. Cet appareil peut servir à recevoir les résultats ve-

nant de l'ordinateur (=output) ou à y introduire des informations (=input). Pour les micro-ordinateurs, il s'agit habituellement du disques durs, de lecteurs de disquettes souples ("floppy disks") ou de mémoires flash.

5.- QU'EST QU'UN LANGAGE?

D'un point de vue matériel, un ordinateur est un appareil électronique traitant des ensembles de signaux électriques. Son fonctionnement est contrôlé par un programme qui est une suite d'instructions. Il ne nous serait pas commode d'introduire des instructions et des informations dans l'ordinateur directement sous forme de signaux électriques (bien que cette méthode fut celle utilisée par les premiers programmeurs). Aussi l'ordinateur est-il fourni avec un programme qui traduit en **langage machine** un **langage de programmation** que nous comprenons facilement.

L'humain parle :

- Java
- C
- Pascal
- Basic
- Excel
- Explorer...

Un programme traduit ou interprète les instructions en langage machine

L'ordinateur parle en signaux électriques

Un langage machine, dit de bas niveau, comporte des instructions qui correspondent directement aux ensembles de signaux électriques que l'on peut produire dans l'ordinateur. Un langage de programmation, dit de haut niveau, est facilement compréhensible par un être humain non nécessairement spécialisé en électronique. Le Java ou le C sont de tels langages.

Ainsi, votre dialogue avec l'ordinateur se décompose en les étapes suivantes.
- Vous avez un problème à résoudre.
- Vous analysez le problème en fonction du langage disponible.
- Vous concevez un programme.
- Vous introduisez ce programme dans l'ordinateur au moyen du clavier.
- L'ordinateur exécute votre programme dans son langage.
- L'ordinateur sort les résultats sous la forme que vous avez précisée dans le programme.

C H A P I T R E II

Les grandes dates de l'évolution de l'informatique

1.- RAPPEL HISTORIQUE

Le premier ordinateur digital, l'ENIAC (Electrical Numerical Integrator And Calculator) a été construit aux Etats-Unis pendant la seconde guerre mondiale en vue de résoudre des problèmes de balistique. Il s'agissait d'une énorme machine occupant une salle de 150 mètres carrés, consommant 137kWH et tout juste capable d'effectuer quelques opérations arithmétiques simples. Cette machine avait moins de possibilités que les calculateurs de poche programmables actuels. Le premier “computer” à usages multiples commercialisé fut l'UNIVAC I (Rank Corporation) suivi de près par les premières machines d'I.B.M. (International Bussiness Machine) qui monopolisèrent rapidement le marché, marché qui était considéré en 1954 comme ne devant pas dépasser quelques dizaines de machines.

Les ordinateurs, dits de la première génération, fonctionnaient grâce à des lampes à vide, consommaient énormément d'énergie pour le chauffage des filaments et le refroidissement de l'ensemble.
En 1959 apparut la seconde génération d'ordinateurs utilisant des circuits imprimés à transistors qui rendirent les ordinateurs plus petits et moins chers.
En 1965, le marché comptait environ trente mille machines. A cette date apparut la troisième génération d'ordinateurs (la série SPECTRA 70 de R.C.A.) utilisant des circuits intégrés. Cela réduisait encore le coût du matériel et sa taille tout en augmentant ses performances. L'informatique n'est plus l'apanage des grosses administrations ou des banques mais commence à envahir la plupart des services comptables des entreprises. L'apparition des mini-ordinateurs (DIGITAL Equipment) augmente encore ce phénomène et permet d'inclure des ordinateurs dans des processus de contrôle ou de fabrication. Les ordinateurs trouvent leur place dans les laboratoires et les usines et ne sont plus utilisés exclusivement pour faire des calculs. Ils deviennent de plus en plus performants.

Pour amortir leur coût élevé de fonctionnement, on établit les techniques de travail en temps partagé (=Time sharing), qui permettent à un grand nombre d'utilisateurs disposant de terminaux d'accéder à la même machine sans être (en principe) gênés par les activités de leurs collègues.

En 1974, la technologie de fabrication des circuits électroniques a tellement progressé qu'il est possible de loger sur une “puce” (=chips) toutes les fonctions fondamentales d'un ordinateur. Le microprocesseur est né et va révolutionner l'informatique en

permettant le développement de la quatrième génération d'ordinateurs. Cependant l'inexistence de programmes spécifiques fonctionnant sur ces machines explique que, au départ, seuls les “hobbyists” reconnaissent la valeur de ces nouveaux circuits et jouent à les programmer. En 1977, plusieurs firmes, récemment créées par des électroniciens enthousiastes, distribuent des machines appelées micro-ordinateurs, dont le faible prix les rend accessibles à tout un chacun. L'informatique individuelle va rapidement offrir une alternative à l'informatique centralisée, concentrée autour de machines de plus en plus grosses, dont le contrôle est entre les mains de quelques programmeurs. Il faut attendre 1981 pour que les géants traditionnels de l'informatique reconnaissent l'utilité des micro-ordinateurs et que, poussés par le déferlement des "micros", ils s'intéressent activement à ce marché.

Pendant toutes les années quatre-vingt, le processus s’amplifie et la micro-informatique déferle sur le monde. Le marché devient très important et une firme de logiciel (Microsoft) s’impose par ses produits. Les applications de bureautique (tableur et traitement de texte) commencent à être utilisées partout.

Les années nonante vont voir un nouveau phénomène prendre corps, celui de la mise en **réseau** des machines individuelles. Le “Personal Computer” s’intègre dans la toile d’araignée mondiale tissée par les sociétés de télécommunication.

En 1999, le nombre d'ordinateurs dans le monde est de l'ordre de 250 à 300 millions de machines. Il s'agit principalement de PC (Personal Computers). La part de marché occupée par les "gros" ordinateurs (dit "mainframe") diminue constamment bien que quelques modèles de "super-computers" soient encore en construction (cf. certaines machines d’IBM et les Crays par exemple).
Le parc des machines se répartit environ comme suit.
• 200-300 millions de machines Intel (dont ≈90% utilisent WINDOWS), La plupart sont construites par de petites sociétés situées en Asie. Un grand nombre de ces micro-ordinateurs sont domestiques (>200 millions) et sont utilisés aussi comme support pour les jeux vidéo.
• 30-40 millions de Macintosh d'Apple,
• 4-6 millions de stations de travail de haut niveau fonctionnant habituellement sous le système d'exploitation UNIX.
• Quelques centaines de milliers de "gros ordinateurs".
En ce qui concerne les logiciels et les systèmes d'exploitation, le système Windows de Microsoft est le plus représenté (85-90%). Le reste du marché se répartit entre le système Apple (Mac-OS) et le système LINUX. Notons que ces systèmes reposent de plus en plus sur des interfaces graphiques (GUI - Graphical User Interface) qui simplifient la présentation et la gestion des machines. La plupart des logiciels adoptent des règles de fonctionnement semblables et sont d'ailleurs développés simultanément pour différents systèmes d’exploitation (cf. Excel, Word, etc…). Ceci permet aux utilisateurs de s'adapter rapidement à différents environnements et facilite la mise en oeuvre des nouveaux programmes.

2.- LES DATES IMPORTANTES

2.1.- Préhistoire

Blaise Pascal (1623-1662) fabrique à 18 ans la première machine à additionner. Charles Babbage (1792-1871) construit le premier automate comprenant une unité de calcul programmable.

2.2.- Première génération

1939 Von Neumann et ses collègues définissent les fondements mathématiques de l'ordinateur. Il s'agit d'un système composé de deux parties : une unité logique et arithmétique capable d'effectuer un nombre restreint d'opérations fondamentales et une mémoire qui contient le programme et les données. Le programme décrit la façon dont les opérations fondamentales doivent être chaînées pour modifier d'une façon importante les données. Les deux idées neuves sont :
- la finalité de la machine n'est pas connue lors de sa fabrication,
- le programme et les données sont placés sur un pied d'égalité en ce qui concerne leur représentation.

1944 Première machine à calculer électronique, l'ENIAC (Electronic Numerical Integrator and Calculator) développée à l'université de Pennsylvanie pour les calculs de trajectoires d'obus (gros succès en physique mathématique). Les programmeurs sont des génies du fil électrique et câblent chaque opération sur des panneaux de contrôle. Cette machine de la première génération fonctionne avec des tubes à vide et est refroidie par l'un des plus gros systèmes de réfrigération jamais construit. Il faut sans arrêt changer les tubes défectueux.

1950 L'UNIVAC 1 de Rank Corporation est le premier ordinateur commercial.

1954 IBM (International Business Machine) entre dans le marché des ordinateurs que l'on supposait ne pas devoir dépasser les 100 machines.

1956 Description du premier langage évolué, le FORTRAN, qui permet aux scientifiques de développer eux-mêmes leurs programmes.

2.3.- Deuxième génération

1959 On passe à la deuxième génération de machines : circuits imprimés et transistors. De nouveaux langages apparaissent : le COBOL (défini par l'administration américaine), l'ALGOL60 (premier langage structuré), le BASIC (destiné à l'initiation au FORTRAN) et le LISP destiné aux recherches dans le domaine de l'intelligence artificielle. Les cartes perforées sont reines.

1960 Les gros projets concernant la recherche spatiale reçoivent des budgets énormes et vont devenir des pôles de développement pour l'électronique et l'infor-

matique. Les banques et les administrations passent au "cerveau électronique" pour leur gestion. On commence à faire fonctionner les ordinateurs en "batch processing" (traitement par lots). Ceci augmente le rendement de ces machines horriblement coûteuses, manipulées par des "gourous" (généralement formés par IBM) au salaire de plus en plus élevé.

1963 Les mini-ordinateurs apparaissent. Il s'agit d'ordinateurs destinés à assurer des tâches spécifiques et pouvant être incorporés dans des systèmes (avions, chaînes de montages etc...). Les firmes Digital Equipment et Hewlett-Packard occuperont ce marché, négligé par IBM.

1965 Les ordinateurs peuvent travailler en temps réel, c'est-à-dire orienter leurs programmes en fonction de stimuli extérieurs. On introduit le "time sharing" (temps partagé), c'est-à-dire que la même machine peut dialoguer en même temps avec plusieurs utilisateurs assis devant des terminaux.

2.4.- Troisième génération

1966 Les circuits intégrés sont utilisés pour construire des ordinateurs de la troisième génération, plus fiables et moins chers (série SPECTRA 70 de RCA). Hewlett-Packard fabrique le HP35, ancêtre de toutes les calculatrices de poche.

1968 Les langages évoluent. Niklaus Wirth définit le Pascal. Les compilateurs PL/1 et ALGOL68 sont disponibles. IBM a des problèmes de monopole. Les premières montres électroniques sont mises sur le marché. Les retombées technologiques de la recherche spatiale se font sentir.

1972 La micro-électronique permet de loger sur un seul circuit un très grand nombre de composants. On développe des circuits spécifiques destinés aux terminaux d'ordinateurs (qui reçoivent le vocable "d'intelligents") et aussi aux jeux vidéo.

1974 Intel construit (presque par hasard) un circuit sans usage bien défini (le premier microprocesseur) et ne voit pas de marché pour cet objet. Le système d'exploitation des gros ordinateurs se complique à outrance. Beaucoup d'informaticiens développent des langages ou des systèmes : naissance du langage "C" et de UNIX.

1975 IBM rêve du "Future system", un super ordinateur ultra rapide et ultra-complexe qui couvrirait le monde entier (le gros chaudron de B. Lussato!). Les ordinateurs travaillent en "mémoire virtuelle". Le marché mondial se compose d'environ 250.000 machines.

2.5.- Quatrième génération

1976 Le premier micro-ordinateur individuel de la quatrième génération est mis sur le marché par la firme Altair, il s'agit d'un kit destiné aux “hobbyists”. Contre toute

attente, il a un gros succès commercial, principalement dans les rangs des électroniciens dont plusieurs sont réduits à l'inactivité par la crise et l'abandon de plusieurs projets spatiaux.

1977 Les micro-ordinateurs sont produits en grand nombre par de petites firmes et par Tandy. La firme Apple fondée par S. Job et S. Wozniac à San Francisco à partir du capital obtenu en vendant une vieille camionnette VW et une calculette HP35 a un certain succès.

1978 Les ventes de micro-ordinateurs progressent. Des programmes spécifiques comme Visicalc sont utilisés sur des micro-machines par les professionnels de la gestion.

1980 Le phénomène de l'informatique individuelle est reconnu par les médias et modifie de façon importante les sentiments du grand public vis-à-vis de l'ordinateur. L'informatique entre à l'école (langage LOGO) et n'est plus l'apanage exclusif des informaticiens. IBM abandonne son projet de "Future system". De gros ordinateurs super performants ("number cruncher") sont fabriqués (CRAY-1, CDC CYBER 205).

1982 Les grands de l'informatique (IBM et Digital) entrent dans l'arène de la micro-informatique et revoient leur position vis-à-vis des systèmes intermédiaires. De nouveaux microprocesseurs très puissants sont disponibles (Intel 8086, MC68000, etc...). Apple Computer Inc. occupe la 411ème place dans les "FORTUNE 500". Le langage Ada est défini par le DOD (Département de la Défense Américain). (Ada était le prénom de Lady Lovelace, fille de Lord Byron, assistante de Babbage et premier programmeur).

1983 Les micro-ordinateurs performants de faibles prix (Commodore 64, Apple //, ZX81, etc...) sont vendus à des millions d'exemplaires. Le langage BASIC est roi. Les fabricants évoluent vers la notion de réseaux (style Ethernet) où tous les postes de travail (les micro-ordinateurs) seraient connectés entre eux et auraient accès à des serveurs (grosses machines) qui serviraient de réservoirs de données. L'influence du langage Smalltalk (conçu au PARC -Palo Alto Research Center) est importante. On parle aussi de gros progrès dans le domaine de l'intelligence artificielle.

1984 Il ne faut plus savoir programmer pour utiliser un ordinateur. L' “interface utilisateur”, c'est-à-dire la façon dont la machine interagit avec son utilisateur subit une profonde évolution. Une part importante de la puissance de traitement est utilisée pour faciliter la relation utilisateur/machine. Les programmes dialoguent ("fenêtres", "souris", etc…) et les machines se banalisent (traitement de texte, tableur, gestion de données, etc…). Apparition du premier micro-ordinateur exclusivement graphique fonctionnant avec une souris : le Macintosh d'Apple Computer.

1985 L'évolution explosive de la micro-informatique touche à sa fin. L'ordinateur devient un outil qui s'intègre de mieux en mieux dans presque toutes les activités professionnelles. Les copies d'IBM-PC venues d'Asie inondent le marché. Les prix des périphériques (disque dur, imprimante, etc…) diminuent. La firme Microsoft devient de plus en plus incontournable. C'est la première fois qu'une société vendant exclusivement du logiciel prend une envergure mondiale.

1986 L'accent est mis sur les connexions en réseaux. Des configurations comportant plusieurs postes de travail reliés entre eux par un réseau sont de plus en plus courantes. L'ordinateur individuel parle avec ses frères… Le disque laser pour ordinateur permettra de disposer sur chaque machine d'une immense quantité d'informations.

1987 La puissance des processeurs couplés à des mémoires immenses oblige les constructeurs à repenser les problèmes. Il faut imaginer de nouveaux modes de travail pour rentabiliser harmonieusement ce potentiel… (cf. Windows de Microsoft, OS/2 d'IBM, HyperCard d'Apple, etc…).

1990 La notion d'ordinateur personnel évolue. On utilise de plus en plus des stations de travail qui sont reliées entre elles et à divers périphériques. L'utilisateur, sans vraiment s'en rendre compte, dispose de ressources disséminées et partagées qu'il peut atteindre et activer à partir de sa machine. L'accent est mis sur les applications dites Multimedia qui incorporent l'image, le son et la vidéo aux applications traditionnelles.
Les nouveaux processeurs utilisant soit la technologie RISC (reduced instruction set) soit la technologie CISC (complex instruction set) gomment les distinctions entre les grosses machines et les PC. L'interface graphique utilisant des menus, des fenêtres et une souris (cf. Windows 3 (MS-DOS), X-Window (UNIX) et Macintosh), devient la règle.

1992 Le marché des PC subit une contraction. Suite à la concurrence effrénée induite par une surproduction de machines issues des régions asiatiques (Taïwan, Corée, etc…), les prix diminuent, ce qui impose à plusieurs fabricants européens et américains de réduire leurs marges bénéficiaires en s'adaptant au marché. Pour la première fois de son existence, IBM enregistre une perte financière importante.

1995 Sur le plan des réseaux, l'INTERNET devient un enjeu de société. On assiste à une croissance extraordinaire du nombre d'ordinateurs connectés (≈30 millions). Suite à la mise au point du World Wide Web (WWW), la consultation et la publication de documents graphiques interactifs sont à la portée de tout un chacun. L'action de la société Netscape qui distribue un "Browser WWW" enregistre une croissance extraordinaire.
L'interface graphique "à la Macintosh" est reprise par Microsoft sous la forme de WINDOWS 95. Ce système d'exploitation, lancé en Août 95 à coups de milliards de

publicité et qui a fait la une des journaux pendant quelques mois, définit le style actuel de fonctionnement de tous les micro-ordinateurs.

1997 Des processeurs de plus en plus puissants voient le jour. Citons la série Pentium de Intel, le PowerPC d'IBM et le ALPHA de Digital. Les fréquences d'horloge passent la barre des 200 MHz.

1998 La guerre des prix sur le marché du matériel fait rage et plus aucun constructeur ne peut se démarquer des autres par ses produits ou ses bénéfices. Au niveau logiciel, les produits de Microsoft sont omniprésents. Quelques concurrents (Netscape pour les browsers, Adobe pour le graphisme par exemple) ainsi que l'état américain lui-même entament une procédure judiciaire basée sur les lois antitrust destinée à évaluer la situation monopolistique de Microsoft.
Les scanners, appareils photos digitaux et les imprimantes couleurs sont de plus en plus utilisés. L'accès à l'Internet se banalise.

1999 Microsoft prévoit la sortie de WINDOW 2000, qui ne sera plus basé sur le DOS des années 80 mais sur le système Window-NT déjà présent sur la plupart des machines Intel de haut de gamme utilisées par les entreprises.
Les petits systèmes portables (hand-held + GSM) et les écrans plats à cristaux liquides sont les deux produits "hots" de l'année.

2001 et après... Le 21ème siècle confirme l'omniprésence de l'ordinateur. L'outil devient incontournable et notre société ne peut plus s'en passer! Notre dépendance vis-à-vis de la machine devient telle que le moindre virus un peu actif peut menacer le bon fonctionnement de pans entiers de l'économie!

C H A P I T R E III

Les langages de l'informatique
utilisés dans les applications scientifiques

1.- INTRODUCTION

Les langages de programmation sont nombreux et variés. Dans ce chapitre, nous allons en évoquer quelques-uns qui, actuellement, sont le plus souvent utilisés dans la programmation scientifique. Il s'agit du FORTRAN, du BASIC, du Pascal, du "C" et de Java.
Nous évoquerons d'abord leurs origines et nous comparerons leurs différents jeux d'instructions et leurs possibilités de structuration de l'information.

2.- ORIGINES

FORTRAN

Ce fut le premier langage destiné à permettre l'écriture de programmes par des non électroniciens. Il était principalement destiné à faciliter le transcodage des formules mathématiques (d'où son nom FORmula TRANslation). Il a été proposé en 1956 comme une alternative à la programmation manuelle des ordinateurs. Sa conception repose sur une approche pragmatique qui met en avant l'efficacité du code plutôt que la qualité conceptuelle du langage. Le FORTRAN standard a été défini en 1966 (FORTRAN 66) et a été revu en 1978 (FORTRAN 77). Récemment, une commission d'experts a défini une nouvelle mouture qui vient de voir le jour (FORTRAN 90…).
Le rôle du FORTRAN dans la communauté scientifique reste important principalement parce qu'il serait impensable d'abandonner les nombreux programmes de qualité, écrits en FORTRAN il y quelques années, et qu'il faut pouvoir continuer à les utiliser et à les entretenir. Cependant le FORTRAN ne permet pas de manipuler élégamment autre chose que des nombres et supporte mal les programmes interactifs. De plus le FORTRAN ne possède pas les qualités requises pour incorporer les idées nouvelles comme la programmation structurée, la récursivité, les pointeurs ou la gestion dynamique de la mémoire.

BASIC

Ce langage (Beginner All purpose Symbolic Instruction Code) a été conçu au Darthmouth College (1967 U.S.A.) pour faciliter l'apprentissage de la programmation

par les étudiants. Les auteurs du BASIC ont construit un langage très facile à implémenter sur les machines de l'époque et qui permettait, pour la première fois, une utilisation interactive.
Le langage eut un succès mitigé jusqu'à l'apparition des micro-ordinateurs. Cependant sous l'impulsion de Bill Gates et sa société Microsoft, qui propose depuis 1975 des versions du BASIC pour tous les PC, le BASIC est devenu la "langua franca" de l'informatique individuelle. En effet, le BASIC permet de réaliser très rapidement de petits programmes et, grâce à un éditeur incorporé et à des instructions système telles que NEW, LIST, OLD, etc…, il isole l'utilisateur du système d'exploitation.
Depuis 1967, la syntaxe du BASIC a fortement évolué. Une révision importante a eu lieu en 1978 et les versions actuelles contiennent de très nombreuses extensions (suppression des numéros de ligne, nouvelles instructions et structuration poussée des données, etc…). En réalité, il n'existe pas un langage BASIC mais des dizaines de dialectes plus ou moins distincts. Les versions récentes rivalisent en possibilités et en performances avec la plupart des autres langages. Ces approches rendent le langage complexe et très spécifique et l'éloignent de son rôle premier qui était de fournir un langage simple destiné à résoudre de petits problèmes.

Pascal

En même temps que naissait le FORTRAN, un comité d'experts tentait de définir un nouveau langage qui, tout en restant simple et efficace, inclurait les notions importantes de structuration des algorithmes (for, while, case, etc…) et des données (array, record, pointeur, etc…). De plus ce nouveau langage devait permettre de composer des programmes faciles à relire et à expliquer. Ce sont ces concepts qui ont donné naissance à l'ALGOL dès 1960 et ensuite à toute une génération de langages structurés dont le plus représentatif, défini en 1972 par N. Wirtz, est le Pascal.
L'intérêt pour le Pascal s'est surtout manifesté dans les années quatre-vingt lorsque plusieurs universités l'ont adopté comme langage d'initiation. Depuis, il est devenu le langage dominant dans l'enseignement. Ce mouvement s'explique par le besoin de disposer d'un langage simple, possédant un ensemble cohérent d'instructions permettant de structurer à la fois les programmes et les données. Cette approche favorise l'adoption par les étudiants d'un style de programmation élégant et efficace.

Le "C"

Historiquement, le "C" fut créé en 1978 au laboratoire Bell pour implémenter le système d'exploitation UNIX sur des machines de la gamme PDP-11 de DIGITAL. La force du "C" réside dans le fait qu'il allie les avantages d'un langage structuré de haut niveau comme le Pascal à l'efficacité du langage assembleur. Il permet d'écrire des programmes extrêmement puissants et compacts. Actuellement le "C" subit de nombreuses extensions et peut être considéré comme un langage universel. Il reste néanmoins un langage pour les programmeurs chevronnés plutôt que pour les novices. Notons aussi que le texte d'un programme écrit en "C" est habituellement assez difficile à comprendre et que l'absence de vérification de la cohérence des expres-

sions (pas de "type checking") peut conduire, dans des mains inexpérimentées, à des programmes dont le fonctionnement est incertain. Le "C" a maintenant un successeur direct qui est le langage C++. Le C++ reprend toute la syntaxe et les instructions du "C" mais y ajoute des notions nouvelles issues de la programmation objet-orientée (cf §7 ci-dessous).

JAVA

JAVA est un langage informatique récent qui intègre les nouvelles possibilités de programmation offertes par le réseau Internet. JAVA permet d'écrire des programmes téléchargeables (Applets) qui fonctionnent directement sur la plupart des ordinateurs.
JAVA repose sur les techniques de programmation les plus modernes. Il est complètement objet-orienté et inclut, entre autres, des mécanismes permettant la programmation en parallèle (threat) et la gestion des erreurs (exception). Le langage peut être étendu dans toutes les directions grâce au mécanisme des classes externes qui permet d'inclure dans un programme des "objets" au comportement spécifique (base de données, multimedia, etc...).

3.- COMPARAISON DES INSTRUCTIONS

3.1.- Ordre d'exécution

Tous les langages considérés exécutent normalement leurs instructions dans un ordre correspondant à l'ordre des lignes du programme (ordre séquentiel). Dès qu'une instruction est terminée, l'instruction suivante du texte est exécutée.
Cependant, tous les langages possèdent des instructions de déroutement qui permettent de changer l'ordre naturel d'exécution pour réaliser des boucles. Ces instructions de déroutement peuvent être conditionnelles. Il est alors malaisé de prévoir l'ordre exact dans lequel les instructions vont s'enchaîner puisque le déroulement du programme dépend de l'évaluation de conditions dont les valeurs changent constamment pendant l'exécution du programme.

Une autre façon de changer l'ordre d'exécution des instructions est de faire appel à des sous-programmes.

Notons que, dans certains langages (Pascal, "C", JAVA), il est possible de regrouper plusieurs instructions sous forme d'une instruction composée. Cette instruction composée est alors formellement assimilée à une seule instruction.

3.2.- Les opérateurs

Tous les langages acceptent évidemment les opérateurs classiques comme +, -, x, etc… Les parenthèses sont utilisées pour changer l'ordre d'évaluation des expres-

sions. Cependant chaque langage possède certaines particularités, parfois subtiles, qui le singularisent.

3.3.- Conditions

Une condition est une expression qui peut prendre deux états (vrai/faux) et qui est utilisée dans les instructions conditionnelles par exemple. Il peut s'agir d'une variable booléenne, d'une expression formée à partir des opérateurs de comparaison (=, >, <, etc…), ou des opérateurs logiques (and, or, etc…). Notons qu'en "C" et en JAVA, une condition est une expression numérique qui est fausse si sa valeur est zéro et vraie lorsque sa valeur est différente de 0. Il est possible de mélanger des opérateurs de comparaison (<, >, !=, ==, etc…) ou logique (&&, ||, !) avec des expressions numériques car le résultat de ces opérateurs est un entier (0 si faux et 1 si vrai).

3.4.- Les instructions fondamentales

Lorsqu'un programme s'exécute, les instructions s'enchaînent normalement séquentiellement selon l'ordre d'écriture. L'ordre d'exécution peut cependant être modifié par des instructions de rupture de séquence. Ces instructions de déroutement sont généralement conditionnelles et leur comportement dépend de la valeur (vraie ou fausse) d'une expression booléenne.
Tous les langages disposent au minimum d'instructions qui permettent de changer l'ordre d'exécution des instructions de calcul et de réaliser des boucles. Certaines de ces instructions sont dites conditionnelles car leur effet dépend de l'évaluation d'une condition.

a) Déroutement inconditionnel (GOTO)

Cette instruction déroute inconditionnellement le programme vers une instruction repérée par un label.
Depuis que Djikstra a banni l'usage du GOTO, la polémique autour de cette instruction n'a pas de cesse. Le problème vient du fait que le GOTO transfère le contrôle du programme vers n'importe quelle autre instruction. Lorsque l'on relit (ou que l'on modifie) un programme "plein de GOTO", chaque label est probablement la cible d'un ou plusieurs GOTO que l'on ne peut pas facilement identifier. Il est donc impossible de mesurer exactement les conséquences de la moindre modification!

b) Déroutement conditionnel

```
if <condition> then <instruction> [else <instruction>];
```

Formellement, l'instruction conditionnelle exécute une instruction si une expression logique est vraie (then) et une autre dans le cas contraire (else). L'option else est souvent facultative. Si elle est absente, rien ne se passe lorsque l'expression booléenne est fausse.

c) Répétition non conditionnelle (for) - Boucles

```
for <var. index> := <depart> to <final> do  <instruction>;
```

Cette instruction permet d'exécuter une ou plusieurs autres instructions un nombre déterminé de fois. Une variable sert de compteur. Les expressions <départ> et <arrivée> peuvent être des constantes, des variables ou des expressions calculées. A chaque itération, le compteur change de valeur et, lorsque la limite est atteinte, l'exécution séquentielle reprend.
Si les bornes sont telles qu'aucune valeur du compteur n'est possible, l'instruction n'est pas exécutée du tout (cependant en BASIC et FORTRAN IV, le corps de la boucle s'exécute toujours au moins une fois).
On peut utiliser le compteur (en lecture) pour suivre la progression, mais il est dangereux de modifier le compteur pendant l'exécution de la boucle. Notons que les instructions de bouclage peuvent être emboîtées ("nested").

d) Instructions de répétition conditionnelle

```
while <condition> do <instruction>
do <instruction>;  while <condition>;
```

Ces deux expressions permettent d'exécuter conditionnellement une instruction. La seconde version, exécute toujours l'instruction au moins une fois avant de tester la condition, tandis que la première n'exécute l'instruction que si, au départ, la condition était vraie.
Notons que pour éviter une boucle sans fin, il faut toujours que l'instruction conditionnelle finisse par modifier la condition logique.

e) Aiguillage selon une valeur (case ou switch)

```
case <selecteur> of
 cas1 : <instruction>;
 cas2 : <instruction>;
 cas3,cas4… : <instruction>;
 …
 [otherwise <instruction>]
end
```

Lorsque le contenu de la variable <selecteur> coïncide avec l'une des valeurs définies pour chacune des rubriques, l'instruction correspondante est exécutée. Attention, en fonction du langage, cette instruction peut avoir un comportement particulier! Par exemple, en "C" et en JAVA, la valeur du sélecteur est comparée successivement à chacune des expressions cas1, …, casn. Dès qu'une égalité a lieu, le programme exécute toutes les instructions qui suivent cette constante, y compris l'éventuelle instruction précédée de "default". Pour éviter ce phénomène et simuler

le "case" du Pascal, il faut terminer chaque instruction (composée) par l'instruction "break".

f) Instructions diverses

Les constructions considérées ci-dessus sont loin de constituer une étude complète. Nous désirons simplement permettre au lecteur de se rendre compte des similitudes de fonctionnement des langages.

4.- LA STRUCTURATION DES DONNEES : LES TYPES

La possibilité de définir soi-même des structures complexes, en fonction de ses besoins, n'existe pas dans tous les langages. Les premiers langages comme le FORTRAN ou le BASIC n'admettent que des structures prédéfinies et n'offrent pas la possibilité d'en construire d'autres. Par contre, certains langages plus modernes comme l'Algol ou le Pascal et le C permettent de concevoir et de manipuler facilement des structures de données très complexes. On parle du type d'une donnée en faisant référence à la façon dont l'information y est codée.

Lorsqu'un langage permet au programmeur de définir ses propres structures de données en plus des algorithmes de traitement, la programmation prend une nouvelle dimension. Concevoir un programme ne consiste plus seulement à coder des instructions mais aussi à définir des structures de données bien adaptées aux problèmes. Une stratégie de travail consiste donc à définir parallèlement la forme de ses objets (les types des données) et les méthodes permettant de les manipuler (les algorithmes).

Parmi les types, une certaine classification peut être réalisée en tenant compte de la façon dont les informations y sont repérées et manipulées. Nous allons tenter de définir les grandes classes auxquelles les types usuels peuvent être raccrochés.

4.1.- Les types simples

Les types simples constituent les bases fondamentales de la représentation de l'information dans un ordinateur. Un type simple associe un nom avec un seul objet élémentaire comme un nombre ou une lettre. Parmi les types simples, on trouve les nombres entiers, les nombres réels, les caractères, les booléens et les pointeurs. Tous les langages considérés acceptent un certain nombre de types prédéfinis. Le type **entier** (nombre entier sans décimale) et le type **réel** (nombre avec décimales) sont présents dans les cinq langages. Notons cependant que la dynamique et la précision de ces types diffèrent habituellement d'une installation à l'autre.

Pour les **entiers,** il existe deux dynamiques possibles :

-32 768 à +32 767 (entiers simples 2 octets)

-2 147 483 648 à +2 147 483 647 (entiers longs 4 octets).

Le seul problème que l'on peut rencontrer dans les calculs faisant intervenir des entiers provient des dépassements de capacité. Certaines versions des langages ne détectent pas l'erreur et rendent parfois des résultats étonnants lorsqu'un résultat

(même intermédiaire) dépasse la capacité dynamique du nombre. Par exemple, 20000*2 rend un résultat négatif!

Pour les **réels**, les choses sont plus délicates. La "simple précision" permet généralement d'effectuer les calculs avec 5 à 6 chiffres significatifs dans une dynamique de $\approx 10^{-30}$ à $\approx 10^{+30}$. Mais il existe bien d'autres représentations pour les nombres réels (double, triple, quadruple, extended, etc…). Leurs caractéristiques dépendent de l'installation. Par exemple, la précision étendue ("extended") des coprocesseurs arithmétiques récents permet de travailler sur une dynamique de $\approx 10^{-4096}$ à 10^{+4096} avec 18 à 20 chiffres significatifs…
Parmi les réels, la valeur 0.0 est toujours disponible et jouit d'un statut spécial. De plus, certaines implémentations acceptent de traiter les "infinis" (comme une division par zéro) avec plus ou moins de bonheur.
Attention, les problèmes liés aux "erreurs d'arrondis" sont souvent très délicats à maîtriser et il faut noter que ce n'est pas tant la taille de la machine qui importe mais surtout la qualité des routines de calculs associées à la version du langage utilisé.

Tous les langages connaissent le type **caractère**. Une variable de ce type peut contenir un symbole (lettre, chiffre, etc…) et être utilisée pour traiter les informations textuelles.

Le type **booléen** (LOGICAL en FORTRAN) peut prendre seulement deux valeurs, vrai et faux, et dispose d'opérateurs spécifiques qui interviennent dans la construction d' expressions logiques.

Le "C" et le Pascal permettent aussi de déclarer et d'utiliser des variables de type **pointeur**. Un pointeur est une variable dont le contenu est l'adresse d'une autre variable. Ce type d'objet est primordial pour gérer l'occupation dynamique de l'espace mémoire et pour créer des structures particulières comme les listes, les arbres ou les graphes.
Notons que l'utilisation des pointeurs (et d'un pointeur d'un pointeur ou "handle") est généralisée dans les applications faisant appel aux ressources des systèmes d'exploitation modernes (Windows de Microsoft, ou Mac Toolbox d'Apple par exemple…). De plus, l'accès et le contrôle des périphériques sont souvent facilités par une utilisation adéquate des pointeurs. Le Java, qui n'accepte pas de gérer des pointeurs, fournit des mécanismes puissants qui permettent de pallier à ce manque.

4.2.- Constructeurs de types

a) Le tableau ("array")
Tous les langages disposent d'instructions qui permettent de construire des tableaux à une ou plusieurs dimensions. Un tableau regroupe sous un même nom plusieurs variables de même type. Chacune de ces variables est repérée par un ou plusieurs indices (entiers) qui peuvent être obtenus comme résultats d'un calcul.

b) Chaînes de caractères (string)

Les langages permettent aussi d'utiliser des constantes et des variables qui contiennent des **chaînes de caractères.**

En pratique les chaînes de caractères sont des tableaux (array) dont les éléments sont des caractères.

Les variables ainsi déclarées jouissent d'un statut spécial : on dispose habituellement d'un grand nombre de fonctions et opérateurs pré-déclarés, soit comme extensions du langage (java, Pascal et BASIC), soit sous forme de librairie de sous-routines ("C" et FORTRAN), qui facilitent la gestion de ces variables.

Les constantes chaînes de caractères sont définies par le texte compris entre des guillemets ("Toto"), ou des apostrophes ('Toto'). Les constantes chaînes de caractères sont souvent utilisées dans les instructions d'écriture pour envoyer des messages à l'écran ou sur papier.

c) Enregistrements (record)

Le Pascal et le "C" permettent au programmeur de construire des types nouveaux grâce aux constructeurs **`RECORD`** et **`STRUC`** qui associent des élements de types différents sous un même nom, ce qui forme ce que l'on appelle un enregistrement. Par exemple, si l'on désire associer sous une même variable le nom (chaîne de caractère), l'âge (entier) et la taille (réel) d'une personne, on construira le type nouveau "personne" qui regroupera ces éléments dans un seul type.

Chaque rubrique d'un enregistrement s'appelle un champ.

Dès qu'un nouveau type est défini, on peut créer, à volonté et n'importe où dans le programme, des variables appartenant à ce type par simple déclaration. Dans le programme, on peut référencer les "champs" des variables de type `personne` en utilisant le nom de la variable suivi du nom du champ précédé d'un point.

Les variables de type enregistrement peuvent être considérées comme des entités qui peuvent être manipulées comme un tout.

Lorsque l'on peut définir des enregistrements dont certains champs sont des pointeurs, des tableaux, ou même d'autres enregistrements, (préalablement déclarés), il devient possible de construire des objets extrêmement complexes dont le bon usage impose une nouvelle approche de la programmation. Il ne suffit plus de programmer ses algorithmes, mais il faut aussi choisir la meilleure structure pour chacun des objets que l'on va manipuler. Cette réflexion est la base de la programmation dite "objet-orientée" et qui privilégie l'aspect "définition des types" par rapport à la programmation des algorithmes.

Notons que le FORTRAN et le BASIC, qui ne contiennent pas de constructeurs permettant d'associer sous un même vocable des objets de types différents, ne rencontrent pas ces préoccupations.

Notons que les langages considérés ne permettent généralement pas de définir des variables dont le contenu serait un "programme" (ou tout au moins une fonction ou une procédure). Cette lacune est en partie comblée par certaines implémentations récentes qui ouvrent des possibilités importantes dans le domaine des programmes "auto-modifiants"(cf. par exemple les "Objets" du Pascal et le C++).

5.- STRUCTURE DES PROGRAMMES

Les instructions de structuration permettent de décomposer un gros programme en petits modules indépendants qui seront imbriqués les uns dans les autres selon des règles bien précises. La possibilité de décomposer une tâche en plusieurs modules est une caractéristique importante des langages structurés, qui simplifie le codage d'un problème en autorisant le développement indépendant de solutions partielles susceptibles d'être regroupées.

5.1.- Structuration en "blocs" (procédure, fonction, etc…)

Les structures formées par procedure et function forment ce que l'on appelle des **blocs**. Un bloc est activé lorsque son nom apparaît dans une instruction. Dans certains cas, une procédure ou une fonction peut s'appeler elle-même : c'est la récursivité. Parfois, un bloc peut contenir des définitions d'autres blocs pour créer des structures emboîtées.

La définition d'une procédure ou d'une fonction se compose de son nom, d'une liste facultative de paramètres et, pour une fonction, de son type. Viennent ensuite les déclarations et les instructions qui seront exécutées à chaque invocation du bloc.
Une fonction retourne une valeur comme un entier, un réel ou une variable de type énuméré, tandis qu'une procédure ne retourne pas de valeur.
Un bloc peut être déclaré "forward" (en avant). Dans ce cas, il doit être défini plus loin dans le texte du programme. Consultez le manuel pour connaître les subtilités de cette option.
Dans la plupart des implémentations, un bloc peut être déclaré external, ce qui signifie qu'il n'est pas défini dans le corps du programme principal mais qu'il décrit uniquement le mode d'appel d'une procédure externe (souvent écrite dans d'autres langages) qui sera adjointe au programme principal lors de l'édition des liens réalisée par le "linker" ou le "task builder".

Les paramètres peuvent être transmis aux blocs, soit par référence (on transmet alors uniquement un pointeur vers l'original), soit par valeur (on transmet alors la valeur du paramètre qui sera recopiée dans une variable locale). La transmission de blocs (procédure ou fonction) sous forme d'arguments est souvent soumise à restrictions et parfois interdite par l'implémentation.

5.2.- Mode de paramétrage des sous-programmes

Nous allons discuter les mécanismes qui gouvernent le transfert d'informations lors de l'appel d'un sous-programme (ou d'une fonction) qui contient une liste de paramètres. Cette liste de paramètres est définie lors de la déclaration du sous-programme ou de la fonction. Les noms de variables qui y apparaissent sont des paramètres formels qui, lors de chaque appel, s'identifient chacun avec les variables correspondantes énumérées dans l'instruction d'appel. Il doit toujours y avoir correspondance biunivoque entre les éléments des deux listes de paramètres, tant en nombre qu'en type.

Il existe trois modes principaux de transfert :

•Transfert par valeur

Le programme appelant donne au sous-programme les valeurs des contenus des variables (ou des expressions) utilisées comme paramètres. Ces valeurs sont alors automatiquement placées dans des variables locales, de mêmes types, que le sous-programme peut utiliser.

•Transfert par adresse (ou par référence)

Le programme appelant donne l'adresse des variables transmises au sous-programme. Grâce à ces adresses, le sous-programme peut accéder aux originaux, soit pour les lire, soit pour les modifier.

•Transfert par double copiage

Lors de l'appel du sous-programme, les contenus des variables paramètres sont transférés dans des variables locales, de mêmes types, et, lorsque l'on quitte le sous-programme, les contenus de ces variables sont automatiquement recopiés dans les variables originales.

Le Pascal soutient explicitement le transfert par valeur et par adresse. Le choix se fait lors de la déclaration de la procédure ou de la fonction. En "C" tous les transferts se font par valeur sauf pour les tableaux qui sont transmis par référence. Lors de l'appel d'une fonction, le contenu de tous les paramètres (sauf tableaux) sont recopiés dans des variables locales fraîchement créées. Cependant, on peut simuler le transfert par adresse en faisant usage de l'opérateur "&" qui, lorsqu'il précède un nom de variable, donne un pointeur vers cette variable. En "C", le choix du mode de transfert se fait lors de l'appel de la fonction.

En "C" il n'y a pas de test sur l'équivalence de type entre les paramètres formels et effectifs. C'est au programmeur de s'assurer de la compatibilité des transferts et il est habituel de profiter de ce laxisme pour réaliser certains "effets spéciaux" qui reposent sur la représentation interne de l'information (qui est spécifique à chaque type d'ordinateur). Cependant, certaines versions du "C" permettent d'inclure dans la liste des paramètres, des informations sur leur type. Ces informations seront utilisées lors de la compilation pour vérifier l'équivalence de type entre les paramètres formels et actuels. Cette pratique s'appelle le "prototyping".

En FORTRAN, le transfert de paramètres se fait (habituellement) par double copiage pour les variables simples et par par référence pour les tableaux.

5.3.- Visibilité des identificateurs

La visibilité ("scope") des identificateurs est déterminée à partir des règles qui régissent la création, la destruction et la préséance des identificateurs dans un programme. Tous les langages considérés utilisent une méthode lexicographique pour établir les règles de visibilité. Cela signifie que c'est la structure du texte du programme (et non pas l'ordre d'appel des procédures lors de l'exécution) qui détermine la visibilité des identificateurs.

En Pascal, en "C" et en JAVA, tous les identificateurs doivent être déclarés explicitement. Lorsqu'une (pseudo) instruction de déclaration est rencontrée, le programme réserve en mémoire une zone dont la taille est fonction du type de la variable. Les variables déclarées en tête du programme sont globales et occupent la même zone tout au long du déroulement du programme. Ces variables peuvent être manipulées à la fois par le programme principal et par toutes les procédures ou fonctions (pour autant qu'une déclaration locale ne masque pas l'identificateur…). Par contre, les zones allouées dans le cadre des déclarations d'une fonction ou d'une procédure sont locales à ce bloc et inconnues ailleurs. De plus, ces zones sont récupérées lorsque l'on quitte cette fonction ou cette procédure (et le contenu des variables locales est irrémédiablement perdu…). On parle d'allocation dynamique de la mémoire. Lors d'appels récursifs, des zones différentes sont attribuées à chaque appel.

En FORTRAN et en BASIC, les variables ne doivent pas être systématiquement déclarées. En FORTRAN 77, toutes les variables qui apparaissent dans un programme se voient attribuer une zone fixe de la mémoire. Toutes les variables sont statiques, il n'y a pas de gestion dynamique de l'occupation. Cependant, les variables déclarées dans le programme principal sont distinctes de celles utilisées dans un sous-programme. Notons que, lors de chaque appel d'un sous-programme, les mêmes zones mémoire sont réutilisées (ce qui empêche les appels récursifs).
Pour faciliter l'échange d'informations entre les sous-programmes, une même zone mémoire peut être atteinte par plusieurs sous-programmes. Il s'agit du mécanisme des "COMMONs" qui est souvent utilisé dans les gros programmes FORTRAN comme moyen de communication entre différents sous-programmes. Cependant, chaque COMMON doit être explicitement (re)déclaré dans chacune des sous-routines qui l'utilisent. Notons que la gestion des COMMONs, qui est de la seule responsabilité du programmeur, est souvent la source d'erreurs difficiles à corriger.

En BASIC, le processus est fort semblable à celui du FORTRAN. Chaque nouvelle variable rencontrée par l'interpréteur se voit allouer une zone mémoire qui est "mise à zéro". Comme toutes les variables sont globales, le problème de l'allocation dynamique ne se pose pas.

6.- INSTRUCTIONS D'ENTREE/SORTIE

Chacun des cinq langages possède évidemment des instructions permettant de gérer des fichiers, d'introduire des données et d'imprimer les résultats. Ces instructions sont complexes et nous n'allons pas les décrire ici. En effet, la spécificité de la gestion des périphériques (disque, imprimante, clavier, etc…) influence subtilement le fonctionnement de ces instructions et en complique la description.

7.- LA PROGRAMMATION "OBJET-ORIENTÉE"

7.1.- Introduction

La programmation OBJET-ORIENTEE est une technique qui permet de structurer un système complexe en le décomposant en **objets** pouvant interagir entre eux en s'envoyant des **messages**. L'objet contient à la fois des données et des algorithmes. Il unifie la représentation des deux éléments de base de la programmation structurée : la structure des données et les algorithmes, en les mettant sur un pied d'égalité formel.

La notion d'objet a été introduite pour la première fois dans les langages SIMULA-67 et Smalltalk-80. L'objet n'est donc pas une nouveauté mais c'est seulement maintenant que l'intérêt pour ce type de structure se généralise. L'"objet" est déjà disponible dans certaines extensions du Pascal (OBJET-Pasca), du "C" (C++ de Bell) ou sous forme de langage nouveau comme Java.

L'intérêt de l'objet réside principalement dans la construction de programmes plus clairs et plus faciles à tester ou à modifier qu'auparavant. La taille et la puissance des ordinateurs actuellement disponibles incitent à la création de programmes complexes, délicats à écrire et à entretenir. Les méthodes de programmation doivent s'adapter et l'introduction de la notion d'objet est un reflet de cette évolution.

7.2.- Définition de l'objet

En C ou en FORTRAN, la composition d'un programme est basée sur la notion de **procédures** ou de sous-programmes, ce qui permet de structurer l'**algorithme** en le décomposant logiquement en sous-modules ayant chacun un rôle simple, facile à décrire et à vérifier.

Le Pascal et le C explicitent la notion de **type**, ce qui permet de structurer les **données** grâce aux RECORDs. On peut, par exemple, introduire le type COMPLEX comme un RECORD contenant deux éléments de type réel (COMPLEX.R et COMPLEX.I). Mais on ne peut pas facilement modifier les opérateurs (+, -, / etc…) pour qu'ils s'adaptent automatiquement au type nouvellement créé. Dans les langages objet-orientés, la notion de type est étendue et contient non seulement la description du contenu des variables mais aussi le fonctionnement des opérateurs (méthodes) qui portent sur ces objets. Par exemple, l'introduction de l'objet COMPLEX permettrait non seulement d'utiliser des variables de type COMPLEX (au sens du Pascal), mais aussi de définir ou de modifier le fonctionnement spécifique des opérateurs portant sur le type nouvellement créé.

Notons que la plupart des langages sont (sur une petite échelle) objet-orientés. Les opérateurs simples (+, -, *) sont généralement "objet dépendants" : la méthode utilisée pour effectuer l'opération est différente selon que l'on traite des entiers ou des réels, mais le choix est automatique, c'est le type des objets qui indique à la ma-

chine la méthode à utiliser. Les langages objet-orientés ne font que généraliser et systématiser ces automatismes à tous les types en les rendant programmables.

7.3.- Un exemple

Voici un exemple qui montre la simplicité conceptuelle de cette approche. Considérons deux objets de types complètement différents, le type PORTE et le type FICHIER. Chacun de ces deux objets peut répondre aux commandes FERMER et OUVRIR. Tout le monde sait ce que veut dire "fermer une porte" ou "fermer un fichier" mais les méthodes sont fondamentalement différentes. Dans les langages informatiques classiques qui privilégient l'instruction, FERMER sera défini en fonction de son action, pas en fonction de l'objet sur lequel l'opération va porter. Si FERMER est défini pour un FICHIER, FERMER(PORTE) ne sera pas valable alors que l'expression est apparemment facile à comprendre. Dans les langages objet-orientés, c'est l'objet qui prime sur l'action. L'instruction FERMER provoquera l'exécution de la méthode adaptée à l'objet sur lequel elle porte.

7.4.- Intérêt de l'approche objet

La simplification de l'approche “objet” plutôt que “action” vient de ce que, en pratique, le nombre de messages (opérateurs) que peut comprendre un objet est restreint et que l'objet a une stabilité temporelle plus grande que l'action. Si nous introduisons les objets DISQUE et PAPIER, le sens des vocables ECRIRE, LIRE, EJECTER est évidente même si les méthodes sont différentes pour chacun des objets. Si nous privilégions l'action, nous aurons du mal à définir, à priori, toutes les méthodes qu'implique l'action ECRIRE.

De plus, chaque fois qu'on introduit un objet, il n'est pas nécessaire de redéfinir toutes les opérations qu'il peut supporter. Un objet peut "hériter" des caractéristiques d'un autre objet préalablement défini. (Par exemple, un chien peut interpréter tous les messages définis pour les mammifères).
Ceci mène aux notions de **classes** et de sous-classes qui font la force des langages objet-orientés. En effet, lors de la définition d'un nouvel objet, on peut signaler qu'il appartient à une classe déjà définie, donc il **hérite** de toutes les propriétés des objets de cette classe. Cet objet sera, de plus, doué de propriétés nouvelles qui définiront sa nouvelle classe (qui sera en fait une sous-classe de la précédente).

7.5.- Concepts nouveaux

Les langages objet-orientés introduisent certains concepts nouveaux reposant sur une terminologie spécifique. Les termes fondamentaux sont : objet, message, classe, instance (exemple) et méthode (la définition de ces termes sera donnée plus loin).
La communication avec l'utilisateur se fait habituellement grâce à un **écran graphique** à haute résolution ("bit map"), un **clavier** et un dispositif de pointage ("**souris**"). L'utilisation du graphisme et la souris sont les caractéristiques les plus apparentes

des systèmes objets, mais pas les plus importantes. Ces interfaces pourront d'ailleurs être remplacées par de plus performantes lorsque cela sera possible (voix, couleurs, etc...).

7.6.- Objet, message, classe, instance et méthode

Le vocabulaire des langages objet-orientés fait constamment référence à cinq mots : objet, message, classe, instance et méthode. Ces différents termes sont définis en fonction les uns des autres (ce qui montre que "le lecteur doit connaître tout avant de comprendre quoi que ce soit"). Ces langages sont orientés vers la manipulation d'objets (contrairement aux langages classiques qui sont principalement composés d'instructions).

Un **objet** est constitué de deux parties, la première contient la valeur de l'objet (il s'agit le plus souvent d'une zone de mémoire) et la deuxième est formée par l'ensemble des opérations que l'on peut appliquer à l'objet (habituellement des programmes).

Une **classe** est constituée par les objets qui ont le même jeu d'opérations mais des valeurs différentes. On dit que ces objets forment chacun une **instance** particulière de la classe.

Un **message** est envoyé à un objet pour activer une opération portant sur une valeur. Le message désigne l'objet, c'est-à-dire qu'il identifie à la fois la valeur et l'opération demandée. Notons que le message ne précise jamais la façon de réaliser l'opération (ceci appartient à l'objet, ou plutôt, à sa classe).

Une **méthode** décrit la façon de réaliser une opération. Les méthodes appartiennent aux objets et ne sont jamais modifiables par les utilisateurs des objets. L'utilisateur ne connaît la méthode que par son nom et ses effets sur la valeur de l'objet. Toutes les instances d'une classe répondent au même ensemble de méthodes.

7.7.- Programmer en objets

Programmer revient à créer des objets et à faire interagir ces objets par des messages. Pour créer un objet, on définit d'abord une classe (en formant un ensemble de méthodes) et on génère l'objet comme une instance de cette classe. Les messages modifient les valeurs d'un objet en exécutant l'une de ses méthodes. Une classe doit contenir une méthode pour chacune des opérations que ses instances peuvent subir. Les méthodes sont elles-mêmes des programmes. Une méthode peut faire référence à d'autres objets appartenant à des classes préalablement définies et envoyer des messages agissant sur ces objets. Une méthode spécifie généralement un objet qui recevra la réponse au message ayant activé la méthode. La notion de sous-classes permet de simplifier le formalisme. Une **sous-classe** forme une nouvelle classe qui **hérite** de toutes les méthodes de la classe mère et à laquelle on ajoute quelques méthodes particulières. Par exemple, pour introduire l'opération d'élévation au cube

dans la classe des nombres, il suffit de créer une sous-classe qui contient les opérations élémentaires (+,-, / etc...) par son statut de sous-classe de la classe des nombres, et qui comprend un message supplémentaire (l'élévation au cube), ce qui la différencie de sa classe mère.

7.8.- L'environnement de programmation

Les langages objet-orientés introduisent habituellement un ensemble de classes qui fournissent les variables et les fonctions habituellement contenues dans un langage de haut niveau. Ces classes permettent de définir et manipuler des objets classiques : nombres et opérations arithmétiques, plume et opérations graphiques, etc... L'utilisateur dispose alors d'un ensemble cohérent d'**outils** qui lui permettent de construire et de modifier ses programmes et qui constituent ce que l'on appelle l'***environnement de programmation***. Ces outils utilisent le **graphisme** et la **souris**. L'usage de **fenêtres** est généralisé. Le système contient souvent des outils qui permettent de visualiser les classes, les objets d'une classe, les méthodes etc... et de modifier interactivement les programmes.

8.- EXEMPLES DE PROGRAMMES

Nous présentons ci-après cinq petits exemples de programmes.

Pascal

```
program somme;
var a,b,resultat:real;

function sum(x,y:real):real;
begin
sum:=x+y;
end;

begin
write("Entrez deux nombres:');
readln(A,B);
resultat:=sum(a,b);
writeln('Leur somme: ',resultat:10:5);
end.
```

"C"

```
#include <stdio.h>

Main()
{
float a,b,resultat;
PRINTF("Entrez deux nombres:");
SCANF("%f,%f",&a,&b);
resultat=sum(a,b);
printf("Leur somme =%f",resultat);
}
```

```
sum(a,b)   /* declaration d'une fonction*/
float a,b;
{
result(a+b);
}
```

BASIC

```
50 INPUT "ENTREZ 2 NOMBRES:" A,B
60 GOSUB 800
100 print "LEUR SOMME:"; RESULTAT

800 RESULTAT=A+B
810 RETURN
200 END
```

FORTRAN

```
      PROGRAM SOMME
      READ(*,10) A,B
      CALL SUM(A,B,RESULTAT)
      PRINT *, RESULTAT
      STOP
10    FORMAT(2F5.2)
      END

     SUBROUTINE SUM(X,Y,S)
     S=X+Y
     RETURN
     END
```

JAVA

```
import java.awt.*;
public class TrivialApplication {
TextField text1;
 public static void main(String args[]) {
 System.out.println( "Hello World!" );
 try {
 System.in.read(); // prevent console window from going away
     }
    catch (java.io.IOException e) {}
 }
}
```

9.- REFERENCES

Comparative programming languages
Wilson L.B, Clark R.G, Addison Wesley (1988)

Pascal user manual and report
Jensen K., Wirtz N., Springer-Verlag (1974)

FORTRAN 77, Approche systématique illustrée d'exemples
Strohmeimer A., Eyrolles (1986)

The C programming language
Kernighan B.W., Ritchie D.M., Prentice Hall (1978)

Pascal from BASIC
Brown P., Addison Wesley (1982)

From Pascal to C
Brown D.L., Wadsworth Publishing Company (1985)

Thinking in JAVA
Eckel B. Prentice HAll (1998)
http://BruceEckel.com

Essential in JAVA
Cowell J., Spinger (1998)

C H A P I T R E IV

Les progiciels

1.- INTRODUCTION

L'utilisation d'un ordinateur implique que l'on mette en oeuvre un ou plusieurs programmes. Ces programmes peuvent être composés par l'utilisateur dans un langage dit de programmation et compilés sur la machine. Cette approche est relativement lourde et implique l'apprentissage d'un ou plusieurs langages très spécifiques. Elle devient de moins en moins nécessaire vu l'apparition de progiciels commerciaux qui permettent de réaliser la plupart des activités que l'on attend d'un ordinateur, sans devoir écrire un programme au sens ancien du terme. Ces logiciels se répartissent en plusieurs catégories comme le traitement de texte, la gestion de bases de données, les tableurs, les présentations graphiques, les "browsers", etc…
Notons que ces programmes permettent souvent le transfert électronique des informations d'une application à l'autre. Par exemple, il est possible de placer, dans un texte, un tableau de nombres construit par le tableur, ou d'utiliser la base de données pour fabriquer des lettres personnalisées. On parle alors de suite de programmes.

Le nombre de progiciels mis à la disposition de l'utilisateur ne cesse d'augmenter. Le choix du programme dépend évidemment de ses besoins et de ses compétences. Un utilisateur peut privilégier la puissance et la complexité ou plutôt la facilité d'utilisation.
Comme pour les livres, il existe des "best sellers". Le marché est actuellement dominé par quelques logiciels commerciaux qui monopolisent le marché. La suite de programmes proposés par Microsoft sous le terme MS-OFFICE (Windows et Mac) est la plus célèbre. La firme ADOBE propose des programmes de dessin et de gestion de documents (pdf). Quark X-Press domine le marché de l'édition. Pour les navigateurs Internet, divers produits se disputent le marché : Internet Explorer de Microsoft, Firefox (Linux et autres), Safari (OS X). D'autres produits occupent des "niches" spécifiques. Dans le domaine de la gestion de données, Access de Microsoft et FILEMAKER d'Apple sont souvent utilisés.

Nous allons décrire brièvement les principales activités que permettent ces programmes. Une description plus technique fait l'objet des notes de travaux pratiques.

2.- TRAITEMENT DE TEXTE

Le traitement de texte aide à la construction de documents destinés à être imprimés. Il fonctionne comme une super machine à écrire qui permettrait de composer, modifier et mettre en page le texte avant de l'imprimer. Il se compose d'un éditeur et d'un formateur. L'**éditeur** permet de visualiser le texte sur l'écran et de le transformer en ajoutant, effaçant ou déplaçant des lettres, des mots ou des paragraphes. Le **formateur** réalise l'impression du texte sur papier et contrôle l'imprimante en fonction des consignes de mise en page (marge, type de frappe, etc…) que l'auteur peut ajouter dans son texte. Le texte (avec ses instructions de mise en page) est conservé sous forme de fichier et peut être repris, retravaillé et imprimé autant de fois qu'on le désire.

3.- TABLEUR ("spreadsheet")

Un tableur permet de constituer à l'écran une feuille de calcul électronique formée de cases contenant soit des **informations** statiques (nombres ou texte) soit des **formules** mathématiques. Le programme calcule en permanence les résultats des formules et les affiche dans les cases correspondantes. Le tableau est toujours “à jour” et réagit immédiatement à chaque changement d'une valeur statique. Les cases et leur contenu sont généralement identifiés par leurs coordonnées cartésiennes (ligne/colonne). Plusieurs astuces de présentation permettent à l'utilisateur de s'orienter et de se déplacer dans le tableau, qui peut avoir des dimensions très importantes (plusieurs centaines de lignes et colonnes). Le tableur permet la visualisation dynamique d'un modèle comptable ou mathématique et constitue un outil puissant de prévision/analyse (what if…?). Le contenu du tableau (valeurs et formules) peut être conservé sur disque ou imprimé sur papier.

4.- GESTIONNAIRE DE BASES DE DONNEES

Un gestionnaire de bases de données manipule des informations encodées sous une forme plus ou moins structurée. Il assure la **saisie** et la mise à jour des données, leur **archivage** sous forme de fichier, et l'**interrogation** et le **classement** selon différents critères imposés par l'utilisateur. Il permet aussi l'impression de **rapports** construits à partir des données initiales. Il doit rendre l'accès à l'information moins cher et plus performant.

Par définition, une base de données est un ensemble organisé d'informations, modifiables et utilisables en vue d'applications particulières, qui prend en compte les interactions logiques entre les informations issues du monde réel.

On peut distinguer trois niveaux d'organisation.
• Philosophique : Principes qui gouvernent les buts et le contenu, et qui sous-tendent la conception et la réalisation générale de la base de données.

• Logique : Définition des clés et des attributs des fichiers ainsi que des relations à établir entre eux (fichiers simples, multiples, relationnels, etc…).
• Physique : Description de la façon de ranger les informations sur les supports mémoire (mémoire vive, disques, …) et des techniques de recherche, de classement, d'affichage etc.
Pour bien faire, il faut assurer la plus grande indépendance possible entre les différents niveaux. Ceci permet, par exemple, de changer de modèle de disque (niveau physique) sans réécrire les programmes de gestion (niveau logique) et de répondre à des questions nouvelles (niveau philosophique) sans changer la structure des fichiers (niveau logique).

Pour décrire la structure logique d'un fichier, on utilise habituellement les termes **fichier**, **enregistrement**, **champs**, **clé** et **attribut** que nous allons brièvement définir ci-après.
Le fichier (“file”) est constitué d'enregistrements (“record”). Chaque enregistrement est subdivisé en champs (“field”) qui contiennent les informations proprement dites. Par analogie, on peut comparer le fichier à un classeur qui contiendrait des formulaires (enregistrements) comportant plusieurs rubriques (champs).
Parmi les différents champs, il en est généralement un qui joue un rôle privilégié: son contenu, appelé “clé”, est utilisé pour classer et retrouver les enregistrements. Dans l'analogie précédente, on peut par exemple classer les formulaires en fonction du nom d'une personne. Ce nom (que l'on trouve dans l'un des champs du formulaire) est alors la clé. Les autres rubriques, qui ne sont pas directement utilisées pour classer les enregistrements, sont appelées “attributs”. Si plusieurs champs sont utilisés pour classer les enregistrements, on parle de clés multiples.
Notons que, en pratique, il est vivement conseillé de choisir comme clé un champ dont la valeur est univoquement définie pour chaque enregistrement (sinon la même clé correspondrait à plusieurs enregistrements, ce qui poserait des problèmes). Ceci explique la propension des administrations à utiliser comme clé un numéro (cf. numéro national) plutôt que le nom.

5.- GRAPHIQUE

Plusieurs programmes permettent de construire facilement des graphiques à partir des données numériques contenues dans un tableur. La présentation peut se faire sous forme d'histogramme (graphique en barres), sous forme de nuage de points, ou même sous forme de "tarte" ou "camembert". Ces logiciels proposent souvent de très nombreuses options. Nous en utiliserons quelques-uns lors des travaux pratiques.
Tous les ordinateurs modernes sont maintenant équipés d’écran graphique. Cette technologie et les différents modes de dessins sont étudiés dans un chapitre spécifique.

6.- MISE EN PAGE ELECTRONIQUE (desktop publishing)

Depuis l'introduction en 1985 des imprimantes à laser bon marché, un micro-ordinateur est capable de produire des documents dont la qualité graphique peut rivaliser avec celle des procédés traditionnels d'édition. Des programmes de mise en page spécifiquement orientés vers la micro édition (Quark XPRESS par exemple) ont révolutionné la composition des textes. L'auteur peut contrôler non seulement le fond mais aussi la forme des documents qu'il conçoit.
La mise en page d'un texte peut généralement se faire directement à l'écran (technique WYSIWYG : What You See Is What You Get) sans compétence particulière (si ce n'est un certain sens artistique…). On peut, par exemple, incorporer des graphiques, des tables et des images dans un texte et changer la taille relative des différents éléments qui composent le document d'une façon interactive et directe. L'impression sur imprimante laser est rapide et fournit immédiatement des épreuves "prêtes à tirer". Le temps qui sépare la révision finale d'un texte de son tirage est réduit. L'auteur peut, jusqu'au dernier moment, reprendre et modifier son document.
Ces nouvelles techniques sont particulièrement utiles pour l'élaboration de documents à petit tirage (rapports internes, notes de cours, thèses de licence, etc.).

6.1.- Un système de mise en page particulier: TeX

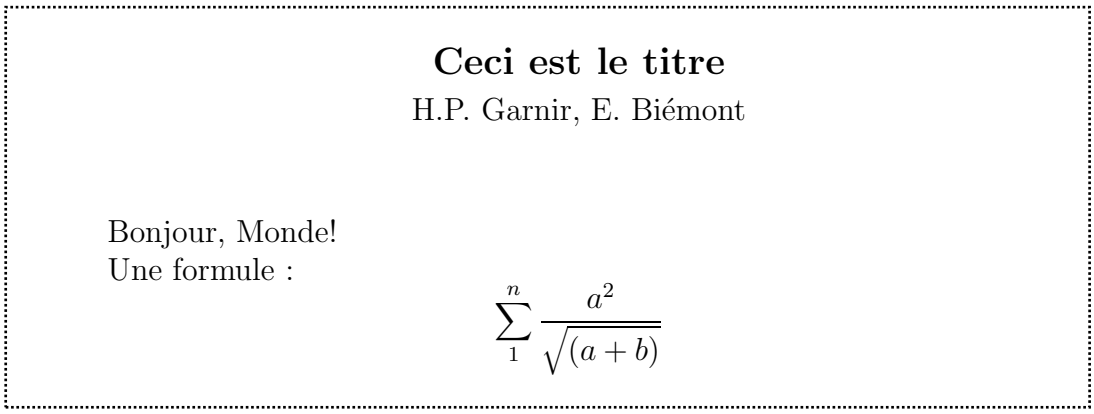
TeX est un outil de compilation de documents. Ce n'est pas un traitement de texte au sens classique du terme. Il permet à l'aide de mots clef spécifiques de dire comment l'on veut voir apparaître son document. La logique qui le caractérise est donnée par les conventions de mise en page. Le système peut alors, tout seul, produire un document le plus convenable possible en utilisant des feuilles de style pour définir la mise en page. Il existe différentes moutures de TeX, la plus souvent utilisée est LaTeX. LATeX est particulièrement étudié pour les mathématiques. Il permet la construction de documents contenant des formules d'une qualité exceptionnelle. On peut, sans trop se tromper, dire que personne ne vient lui faire d'ombre pour tout ce qui touche à l'édition d'équations, et que ses possibilités sont grandes pour tout le reste. LATeX compile des fichiers textuels (ascii ou unicode) standards, avec l'extension .tex. Il reconnaît ses commandes au fait que celles-ci commencent par un backslash (\), et il reconnaît certains caractères comme étant spéciaux. Par exemple, les expressions mathématiques sont encadrées par le sigle $.

Voici un (très petit!) exemple de document LaTeX :

```
\documentclass[french,12pt]{article}
\def\title#1{\begin{centering}\large\bf #1 \\[1mm]\end{centering}}
\def\author#1{\begin{centering}#1 \\[1mm]\end{centering}}
\begin{document}
\title{Ceci est le titre}
\author{H.P. Garnir, E. Bi\'emont}
\vskip 1cm
Bonjour, Monde!
Une formule :
```

```
$$\sum _{1}^{n} \frac{a^2}{\sqrt{(a+b)}}$$
\end{document}
\end{document}
```

Lors de la “compilation” plusieurs fichiers seront créés, parmi eux, un fichier porte un suffixe .pdf C'est le résultat qui peut être visualisé et surtout imprimé. Voici une image du résultat.



Ceci est le titre

H.P. Garnir, E. Biémont

Bonjour, Monde!
Une formule :

$$\sum_{1}^{n} \frac{a^2}{\sqrt{(a+b)}}$$

Notons que le système TeX peut être utilisé gratuitement et que l’American Mathematical Society sponsorise le consortium.

Une adresse utile : http://www.ams.org/tex/public-domain-tex.html

7.- COMMUNICATION

Depuis le début des années 80, des micro-ordinateurs performants de faible prix sont vendus à des millions d'exemplaires. La dispersion des moyens a engendré une évolution explosive de la micro-informatique et a transformé l'ordinateur en un outil qui s'intègre de mieux en mieux dans presque toutes les activités professionnelles. Cependant, la notion d'ordinateur personnel évolue, l'utilisateur désire de plus en plus pouvoir échanger rapidement des informations avec d'autres utilisateurs (proches ou lointains) et accéder rapidement (et simplement) aux nombreuses sources d'informations disséminées à travers le monde. Cette approche implique que les différentes machines soient interconnectées par ce que l'on appelle un réseau.
Dans cette optique, de nombreux postes de travail (les micro-ordinateurs), bien que restant autonomes, sont interconnectés entre eux et ont accès à des serveurs (autres machines) qui jouent le rôle de réservoirs de données. L'utilisateur, sans vraiment s'en rendre compte, dispose de ressources disséminées et partagées qu'il peut atteindre et activer à partir de sa machine.
Nous allons évoquer les grandes lignes de cette évolution.

Un réseau est composé de plusieurs machines qui représentent chacune un ou plusieurs noeuds ("node"). Chaque noeud possède une adresse unique, il s'agit habituellement d'un numéro ("node address") qui permet de l'identifier de manière univoque. Certains systèmes admettent des adresses plus explicites ("node name") en établissant une corrélation locale entre les numéros et les noms.

Le réseau peut couvrir une zone plus ou moins étendue. Si le réseau relie des machines identiques situées physiquement dans le même bâtiment, on parle d'un **réseau local** ("Local Area Network - LAN). Au contraire, si le réseau relie des machines disséminées dans le monde entier, on parle d'un **réseau global** (World Area Network - WAN).

Pour que des machines puissent dialoguer entre elles, elles doivent s'entendre sur un certain nombre de règles qui définissent les modalités de transmission. Ces règles définissent un **protocole** qui doit être implémenté sur chacun des partenaires du réseau. Ces règles sont extrêmement complexes car elles régissent aussi bien l'aspect physique des connecteurs que les concepts qui gèrent les transferts.
Les réseaux feront l'objet d'un chapitre spécifique.

C H A P I T R E V

Le traitement de texte

Un traitement de texte est un outil conçu pour manipuler des textes. Avant d'en donner une présentation plus détaillée, nous rappelons quelques notions qu'il faut avoir en mémoire pour mieux comprendre et donc mieux utiliser ces outils.

1.- STRUCTURE D'UN DOCUMENT

Définir la structure d'un document ne signifie pas définir sa présentation. Sa structure ne dépend que de son sens, sa présentation n'est que l'apparence qu'il prendra. Il y a quatre niveaux conceptuels distincts dans la structure d'un texte (ceux-ci ne ne sont pas directement dépendants du sens) :

1. le document dans son ensemble ;
2. le paragraphe : c'est un bloc, constitué de phrases ; deux paragraphes sont séparés par un retour à la ligne impératif ; un paragraphe peut commencer par un retrait de début de ligne ;
3. la phrase : c'est un groupe de mots (et éventuellement de signes de ponctuation) ; les phrases sont séparées par un point (simple, d'interrogation, ou d'exclamation) ;
4. le mot : c'est un groupe de lettres ; deux mots sont séparés par un espace ou un signe de ponctuation.

Il faut remarquer que la ligne ne fait pas partie de la structure du document. Elle n'apparaît que dans la mise en page.

Lorsqu'on veut écrire un texte, on s'attache souvent, dans un premier temps, à trouver un plan, une structure sémantique. La présentation doit aussi refléter cette structure, qui doit, dans un certain sens, être aussi saisie.

1.1.- Les étapes de la construction d'un document

On peut alors distinguer trois étapes :
1. l'entrée : saisie, mais cela peut aussi être le chargement depuis un disque ; c'est dans cette partie qu'on définit la structure du document ;
2. la mise en page : on associe une présentation à chaque élément de la structure ; en principe, sauf exceptions, tous les éléments de même type doivent avoir la même présentation ; on parle du style de ces éléments ;
3. la sortie : production d'un document papier (impression) ou sauvegarde sur le disque en vue d'une réutilisation ultérieure.

1.2.- Obtention et révision des données

Les textes peuvent être obtenus de plusieurs façons. La frappe au clavier (ou saisie) est la plus directe, mais elle n'est pas la plus adaptée si le document préexiste. Si le document a déjà été saisi, il a certainement été enregistré dans un fichier. On peut alors y avoir accès une nouvelle fois. Si le document est imprimé, on a parfois recours à un scanner et à un logiciel de reconnaissance de caractères. On peut obtenir des textes par le courrier électronique, par une recherche sur Internet, par le biais d'un autre logiciel. Pour généraliser, les textes peuvent provenir d'autres logiciels que Word. On parle alors d'importations de fichiers de divers formats.

Pour permettre une facilité dans la saisie et la révision des textes, le traitement de texte offre des fonctionnalités de déplacement et de défilement du texte, de sélection, de suppression, toujours relatives aux éléments constitutifs de base, à savoir les caractères, les mots, les lignes, les phrases.

1.3.- Modification et vérification

Rechercher tous les Jacques et les remplacer par des Gaston dans votre recueil de poèmes d'adolescent(e)s peut prendre du temps si vous avez aimé Jacques et la poésie. Il est donc intéressant de disposer d'outils qui vous permettent d'effectuer cela sans trop de peine et avec suffisamment de souplesse pour éviter de faire référence à Gaston Prévert en citant vos sources d'inspiration. Ces outils existent dans les traitements de texte.

Les correcteurs orthographique et grammatical semblent de plus en plus courants même s'ils sont souvent peu performants. La correction grammaticale en particulier est délicate à mettre en oeuvre (surtout en français!).

1.4.- Fonctionnalités au niveau du document

Pour certains documents, il est intéressant de pouvoir automatiser la construction de glossaires (un mini-dictionnaire qui donne la définition de mots techniques ou inhabituels), d'index (une table qui donne le numéro de la page où un mot, une notion est utilisé), de tables de matières ou de figures, et de contrôler la pagination, la numérotation, le colonnage, etc.

Permettre des vues de niveaux différents sur son document est important pour l'homogénéité et la continuité. Le plan vous montre le document en suivant le niveau des titres, cache ou découvre des parties si nécessaire.

1.5.- Fonctionnalités liées à la diffusion

On peut imprimer l'information, mais, avec la multiplication des réseaux, on peut maintenant télécopier, émettre un courrier électronique, comme on envoie une note de service ou un document à lire et à commenter. Les traitements de texte se sont adaptés en offrant des fonctionnalités comme, par exemple, le routage, le commentaire, l'équivalent du coup de stabilo ou de la note en marge.

Des lettres types, prises comme modèles de document, automatisent les courriers d'entreprise en inscrivant de base le logo, les adresses, les références, la formule de politesse, etc. Le publipostage permet avec un seul texte et une liste de valeurs (par exemple des noms et des adresses) de créer des dizaines de lettres personnalisées. Les sociétés de marketing direct qui vous annoncent chaque semaine que vous avez gagné une voiture, une montre en or ou un voyage aux Caraïbes se servent beaucoup de cette fonctionnalité. D'autant plus que les étiquettes à coller sur les enveloppes peuvent être créées dans le même mouvement sans aucun effort supplémentaire.

Enfin, après tout ça, si vous parvenez encore à répéter quotidiennement la même suite de 20 manipulations du clavier et de la souris, c'est que vous n'avez pas encore découvert l'enregistrement de macro-commandes qui permettent justement de remplacer tout cela par un léger click sur un bouton.

2.- LA FORME D'UN TEXTE

Nous allons maintenant discuter plus en détail certains points de la mise en forme d'un texte. Comme nous l'avons vu, un texte se compose de caractères qui forment des paragraphes distribués sur une page.

2.1.- Les caractères

Les caractères sont définis par :
1. la police : helvetica, times, courrier, sans sérif, etc. ;
2. le style : gras, italique ou souligné, etc. ;
3. la taille : exprimée en points ; une taille de 8 à 12 points est une taille normale.

Notons que les typographes utilisent des unités particulières :
le point qui vaut 1/72 de pouce (1 pouce = 25.4 mm),
le pica qui vaut 12 points.
De plus, la taille des lettres est mesurée en hauteur et, dans la plupart des polices, les lettres ont des largeurs différentes : un i est plus étroit qu'un m.

2.2.- Les paragraphes

Un paragraphe se compose de phrases qui forment plusieurs lignes de texte. Les retours à la ligne à l'intérieur d'un paragraphe ne doivent jamais être introduits au clavier. La machine se charge du découpage. Par contre, pour séparer deux paragraphes, on doit insérer au moins un retour à la ligne, introduit par la touche <Entrée>. Notons que pour introduire des décalages horizontaux, il ne faut jamais recourir à la touche <Espace> mais modifier globalement la description du paragraphe ou utiliser les tabulations.

Les principaux réglages possibles dans la manipulation d'un paragraphe sont :
1. tous les réglages existants sur les caractères (ils s'appliquent alors à tous les caractères du paragraphe) ;
2. la marge gauche, la marge droite et le retrait en début de première ligne ;
3. l'espacement avant, l'espacement après (la hauteur laissée vide avant et après le paragraphe) ;
4. les veuves et orphelins : la façon dont un paragraphe est autorisé à laisser traîner quelques lignes seules en début ou en fin de page, ainsi que la façon dont on doit lier les paragraphes entre eux (par exemple, un titre ne doit pas rester seul en bas d'une page alors que le texte correspondant est en début de page suivante).

Deux cas peuvent se produire lorsqu'on veut donner un certain style (c'est-à-dire une certaine présentation) à un paragraphe :
1. le texte est court ou comporte un seul paragraphe de ce style : on modifie ce paragraphe, et on passe à autre chose ;
2. le texte comporte plusieurs paragraphes de ce style : ne pas les modifier (taille des caractères, espacement, etc.) un par un ; on modifie un des paragraphes devant avoir cette présentation, on enregistre le style dans la liste des styles, et on utilise cette liste pour définir le style des autres paragraphes.
On peut également utiliser les styles prédéfinis. S'ils ne vous plaisent vraiment pas, redéfinissez-les. Un des avantages de cette façon de faire : si vous changez d'avis quant à la présentation de ces paragraphes, il suffira de changer le style et tous les paragraphes de ce style seront changés.

2.3.- Retrait en début de ligne

On peut définir, sur chaque paragraphe, la marge et le retrait en début de première ligne.
La marge est un décalage par rapport au bord gauche du papier de toutes les lignes du paragraphe, sauf la première.
Le retrait en début de première ligne est un décalage par rapport aux autres lignes du paragraphe.

On peut donc définir des textes :

avec retrait négatif :
Ceci est un texte avec retrait négatif en début
de paragraphe. Ceci est un texte avec retrait
négatif en début de paragraphe.

avec retrait positif :
Ceci est un texte avec retrait positif en début
de paragraphe. Ceci est un texte avec retrait
positif en début de paragraphe.

sans retrait :
Ceci est un texte sans retrait en début de paragraphe. Ceci est un texte sans retrait en début de paragraphe.

Pourquoi utiliser les retraits ?
Il peut sembler tentant d'oublier d'utiliser les retraits, en se disant : << cela va aussi bien avec des espaces. >> Eh bien, non ! Cela ne va pas aussi bien. En effet, le traitement de texte calcule le positionnement des caractères de façon à réussir proprement les alignements des fins de lignes. Et il ajuste ce positionnement en jouant sur la largeur des espaces. Ce qui fait que deux lignes différentes, commençant par le même nombre d'espaces, peuvent très bien ne pas commencer au même endroit.

2.4.- L'alignement

Un texte peut être cadré à droite, à gauche, au centre ou complètement justifié. Il s'agit de la façon dont les lignes sont groupées :

Ceci est un alignement à gauche. Ceci est un alignement à gauche. Ceci est un alignement à gauche. Ceci est un alignement à gauche.

Ceci est un alignement à droite. Ceci est un alignement à droite. Ceci est un alignement à droite. Ceci est un alignement à droite. Ceci est un alignement à droite.

Ceci est un alignement centré. Ceci est un alignement centré. Ceci est un alignement centré. Ceci est un alignement centré. Ceci est un alignement centré.

Ceci est un alignement justifié (les lignes du paragraphe sont alignées à gauche et à droite, sauf la dernière). Ceci est un alignement justifié. Ceci est un alignement justifié. Ceci est un alignement justifié.

2.5.- Séparer deux paragraphes

Pour espacer verticalement deux paragraphes, il faut modifier ce que Word appelle l'espacement entre deux paragraphes.
De la même façon que précédemment, il peut sembler plus simple d'utiliser la touche Entrée pour produire des lignes vides qui semblent avoir le même résultat. Eh bien, non! D'abord parce qu'il est difficile d'ajuster finement ces espacements. Ensuite, parce que cela peut vous amener à avoir des lignes vides en début de page.

2.6.- Aligner à l'intérieur d'un paragraphe

(Par exemple pour faire un tableau)
Il suffit de déclarer des tabulations et d'utiliser la touche <Tab> pour passer de l'une à l'autre.

3.- QUELQUES REGLES DE PRÉSENTATION

3.1.- Quelques conseils généraux

1. N'abusez pas des présentations “exotiques” : chacun a sa propre conception du beau, qui ne correspond pas forcément à la conception généralement admise du lisible. Votre document doit d'abord être lisible.
2. Dans un texte, ne pas mettre une partie à la fois en italique et entre guillemets.
3. La possibilité de manipuler de nombreuses polices de caractères est un avantage des traitements de texte modernes qui devient vite un inconvénient si on n'applique pas la célèbre formule : A consommer avec modération. En effet, n'oubliez pas, une fois de plus, que votre document doit d'ABORD ÊTRE LISIBLE : la multiplication des polices différentes n'améliore pas forcément la lisibilité.

3.2.- Règles de typographie

Rappelons ici quelques règles de saisie usuelles pratiquées en dactylographie française :

L’espace

vous devez placer un espace :

• avant la double ponctuation (! ? : ;) ;

• avant le % et en général les unités monétaires, kilométriques, etc. ;

• avant les guillemets fermants (>>) et le tiret (-) lorsqu'il est utilisé en milieu de phrase ; après >> (et bien sûr après , ; : . ! ?) et tiret.

Remarques

Certains traitements de texte savent insérer automatiquement les espaces avant les doubles ponctuations, et il est alors inutile de les taper. Vérifiez sur celui que vous utilisez ; il peut parfois s'agir d'un réglage à faire.

Ces règles ne concernent que la ponctuation française. En anglais, il n'y a jamais d'espace avant la ponctuation.

Les parenthèses

La principale question est : << Comment mettre des espaces ? >>. La règle est simple : les parenthèses doivent être collées à ce qu'elles entourent ; le bloc ainsi obtenu se comporte comme un mot, c'est-à-dire qu'on mettra des espaces avant ou après ce bloc, comme s'il s'agissait d'un seul mot.

Par exemple :

Ceci est un bon exemple (un exemple, dites-vous ?) d'utilisation des parenthèses.

Ceci est un mauvais exemple (un exemple, dites-vous ?) d'utilisation des parenthèses.

Ceci est un mauvais exemple (un exemple, dites-vous ?) d'utilisation des parenthèses.

Ceci est un mauvais exemple (un exemple, dites-vous ?)d'utilisation des parenthèses.

Ceci est un mauvais exemple(un exemple, dites-vous ?) d'utilisation des parenthèses.

Ce qui suit est plus particulièrement lié à la présentation du document mais doit être pris en compte dès la saisie.

Autres points à surveiller

• les nombres ne s'écrivent pas en anglais comme en français : il faut mettre une virgule pour séparer les unités de la partie décimale, exemple : 1,5 km et mettre un blanc insécable pour séparer les tranches de mille comme dans 12 345,678 91 ;
• la mise en évidence d'un texte ne doit jamais se faire en soulignant ;
• les citations et les locutions latines sont mises en italique dans le texte en romain (sauf pour cf., etc. et toutes les locutions francisées comme critérium) ;
• les noms propres se composent en petites capitales (mais le prénom reste en romain) soit par exemple : Donald KNUTH ;
• les lettres des sigles (MET, SNCB...) ne sont pas séparées par un point, sauf lorsque ce sigle est peu connu et s'énonce lettre par lettre ;
• le caractère insécable ne s'utilise que pour corriger une coupure malheureuse ; son utilisation systématique ne facilite pas la relecture et est bien souvent sans raison d'être dans les textes courants ; toutefois, on pourra l'utiliser entre prénom et nom lorsque le prénom est abrégé comme dans D. KNUTH ;
• les majuscules doivent toujours être accentuées ;
• dans les titres, la première lettre du premier mot est en majuscule ; le titre ne se termine pas par un caractère de ponctuation (à l'exception des points d'exclamation et d'interrogation) ;
• dans une énumération simple comme celle-ci (c.-à-d. ne comportant pas plus d'une phrase par élément), vous devez commencer chaque élément par une minuscule et le terminer par un point-virgule (sauf le dernier).

C H A P I T R E VI

Le tableur

Un tableur permet de saisir des données, de faire des traitements sur ces données et de les afficher. Les originalités du tableur sont l'organisation des données et les fonctionnalités de haut niveau mises à disposition de l'utilisateur. Le tableur est très adapté pour manipuler des chiffres, des listes, pour effectuer des calculs, des statistiques. Il est utilisé par les scientifiques pour organiser, traiter et présenter les données. Il permet aussi d'effectuer des synthèses et des simulations.

1.- LES ELEMENTS D'UN TABLEUR

1.1.- Les cellules

Les données (du texte, des nombres, des dates,...) sont stockées dans des cellules. Chaque cellule se trouve à l'intersection d'une ligne et d'une colonne dans une feuille de calcul. Un classeur est un ensemble de feuilles de calcul.

1.2.- Les références

Chaque élément, classeur, feuille de calcul, ligne, colonne, cellule, est désigné par sa référence. La référence permet d'identifier précisément et sans ambiguïté un élément. Par exemple, les lignes portent des numéros comme 1,2,3..., les colonnes portent des lettres A,B,C, … et les cellules sont référencées comme à la bataille navale A1, B6, D3,... La notion de référence est primordiale dans le tableur. Elle est la base de l'élaboration de formules qui effectuent des calculs. On a la possibilité de nommer par un nom clair une ou des cellules et de remplacer ainsi la notation << bataille navale >> assez illisible.

1.3.- Valeurs, formules, fonctions et formats

Chaque cellule peut contenir une valeur. Cette valeur est soit saisie directement par l'utilisateur ou est le résultat d'un calcul exprimé par une formule. Une valeur possède un type et est affichée sous un format.
Les types sont par exemple : des valeurs numériques entières ou décimales, des mots ou des phrases (appelées en informatique chaînes de caractères), des valeurs logiques (VRAI ou FAUX, appelées valeurs booléennes en informatique).
Les formules sont des expressions qui sont évaluées par le tableur et retournent un résultat. Par exemple, A3 + (Cos(B2*7.3)) est une formule (mathématique). Les formules sont bâties avec des fonctions. Dans ce dernier exemple, Cos est une fonc-

tion, mais aussi +, * et -. La facilité d'utilisation du tableur provient du fait qu'on puisse mettre des références de cellules ou d'ensembles de cellules dans les formules. L'expression sera évaluée avec les valeurs contenues dans les cellules A3 et B2.

Les formats sont les attributs des cellules qui permettent de présenter lisiblement les valeurs. On peut citer par exemple, les formats de date, d'heure, de monnaie, de pourcentage, mais aussi gras, souligné, italique, ... Il faut remarquer que l'affichage du contenu d'une cellule peut être différent de la valeur contenue dans cette cellule.

2.- FONCTIONNALITES

Le tableur est à l'origine un outil d'entreprise, surtout développé en vue des tâches de gestion et de bureautique. Son impact a surtout marqué l'informatique de décision (le "What if", prévisions, prospectives, analyses,...).

2.1.- Saisie de données

Saisie directe des contenus des cellules

L'utilisateur tape directement les données dans les cellules du tableur. Ceci suppose un ensemble de commandes pour se déplacer dans la feuille, sélectionner des cellules, saisir, modifier des valeurs ou des formules, ou encore créer des formulaires types et des macros.

Importation de données

Les données peuvent dans certains cas être obtenues directement sous format électronique à partir d'autres programmes (bases de données, traitements de texte...). Les informations peuvent aussi provenir directement de certains instruments de mesures.

2.2.- Stockage des données

Les feuilles et les classeurs peuvent être enregistrés sous forme de fichiers informatiques classiques et être copiés et archivés comme n'importe quel autre document.

2.3.- Calculs

En plus des opérateurs classiques (+, - /, …), les tableurs modernes disposent d'un très grand nombre de fonctionnalités mathématiques, scientifiques, textuelles, financières, statistiques, ... qui peuvent s'appliquer à tous les types de données reconnues par le tableur. On peut ainsi manipuler les nombres, les dates et les heures, les textes, etc… Il existe bon nombre de fonctions prédéfinies qui permettent aisément d'établir des statistiques, de faire des cumuls et de résoudre des problèmes complexes. Certains tableurs permettent même de trouver les valeurs qui optimisent un processus ou qui permettent d'atteindre un résultat.

2.4.- Formules : références et fonctions

Chaque cellule peut recevoir une formule de calcul, l'évaluation de cette formule fournissant une valeur. C'est cette valeur qui apparaît à l'écran. Pour que le tableur comprenne que le contenu d'une cellule est une formule, on utilise la convention suivante : une formule commence toujours par le symbole =. Une formule est un ensemble de calculs effectués à l'aide d'opérateurs (opérateurs arithmétiques usuels + * -..., opérateurs de comparaison < > =..., etc) et de fonctions sur des données (fonctions statistiques, financières, mathématiques...). Les données peuvent être inscrites littéralement dans la formule ou peuvent se trouver dans d'autres cellules.
Le mécanisme permettant d'accéder, à partir d'une cellule, à une valeur se trouvant dans une autre cellule est appelé référence. Nous allons, dans ce paragraphe, présenter les références, puis les fonctions et les formules.

2.5.- Références à une cellule

Chaque ligne d'une feuille de calcul est repérée par un numéro. Chaque colonne est repérée par une lettre (de A à Z, puis de AA à AZ, puis de BA à BZ, etc...). Ceci permet de désigner une cellule par une lettre et un chiffre qui repèrent la colonne et la ligne à l'intersection desquelles elle se trouve.

2.5.1.- Référence relative

Considérons l'exemple suivant : on désire disposer 4 multiplications sur la feuille de calcul de la façon suivante, les produits étant calculés par Excel grâce à des formules adéquates.

Une première solution consiste à écrire soi-même toutes les formules pour calculer les produits,
ce qui donne :

formule en C6 : =C4 * C5
formule en F6 : =F4 * F5
formule en C10 : =C8 * C9
formule en F10 : =F8 * F9

Autrement dit, il faut taper quatre formules différentes, alors que le principe du calcul est toujours le même : effectuer le produit de la valeur se trouvant deux lignes au-dessus par la valeur de la cellule se trouvant immédiatement au-dessus.

Une autre façon de procéder est cependant possible : écrire la première formule dans la cellule C6. Ensuite on utilise le Copier-Coller. La cellule C6 est copiée puis collée en F6, C10 et F10. Que se passe-t-il lors d'une opération Coller ? A priori, on pourrait s'attendre à ce que la formule soit recopiée telle quelle : on obtiendrait alors la formule =C4*C5 dans la cellule F6, ce qui n'est pas précisément ce que l'on désirait. Ce n'est cependant pas ce qui se passe. La formule obtenue en F6 est bien =F4*F5. Pourquoi ?

Cela vient du fait que les références utilisées lors de l'écriture de la formule dans la cellule C6 sont des références relatives. Ainsi C4 désigne davantage la cellule se trouvant deux lignes plus haut dans la même colonne (et qui se trouve être la cellule C4) que la cellule C4 elle-même. Lorsque cette formule est recopiée dans le presse-papiers, c'est ce point de vue qui est conservé. Lors du collage dans la cellule F6, la référence est modifiée pour continuer à désigner la cellule se trouvant deux lignes plus haut dans la même colonne. Ainsi C4 se transforme en F4 et C5 se transforme en F5, etc... Les formules obtenues sont bien celles que l'on espérait !

2.5.2.- Référence absolue

Considérons maintenant l'exemple suivant : on désire écrire la table de multiplication du nombre se trouvant dans la cellule E2.
De nouveau, les formules que l'on veut copier dans les cellules B5 à K5 sont toutes du même type : faire le produit du contenu de la cellule E2 par le contenu de la cellule immédiatement au-dessus. On va donc écrire cette formule une seule fois dans la cellule B5, puis la recopier dans les cellules suivantes à l'aide du Copier-Coller.
Si l'on s'en tient à ce que l'on connaît pour l'instant, on est tenté d'écrire simplement la formule =E2*B4 dans la cellule B5. Que se passe-t-il si on recopie cette formule dans la cellule voisine ?
Les références utilisées sont des références relatives, donc lors de la recopie, ces références sont modifiées pour désigner toujours la même position relative de cellule.
Ainsi, en B5, B4 désigne en réalité la cellule se trouvant à la ligne précédente dans la même colonne. Après Copier-Coller dans la cellule voisine, la référence est modifiée pour désigner à nouveau la cellule de la ligne précédente et de la même colonne, ce qui donne C4. C'est bien ce que l'on voulait obtenir.
Examinons maintenant le cas de la référence E2 utilisée dans la formule en B5. Elle désigne en réalité la cellule se trouvant trois lignes plus haut et trois colonnes à droite. Après Couper-Coller de la formule en C5, cette référence est modifiée pour désigner la cellule se trouvant dans la même position relative, à savoir la cellule F2 !
Catastrophe ! Ce n'est pas du tout ce que l'on voulait obtenir ! Comment s'en sortir ?
Dans la cellule B5, il faut utiliser une référence absolue sur la cellule E2. Une telle référence s'obtient en faisant précéder la lettre de la colonne et le numéro de la ligne par un signe '$' : cela donne l'écriture $E$2. Cette écriture désigne réellement la cellule E2, et non pas la cellule se trouvant trois lignes plus haut et trois colonnes à droite. Lors d'une opération de Couper-Coller, la référence n'est pas modifiée.
Finalement, la bonne formule à écrire dans la cellule en B5 est =$E$2*B4. Après recopie en C5, on obtient la formule =$E$2*C4 (la référence absolue n'est pas modifiée, la référence relative est mise à jour).

2.5.3.- Référence mixte

Une référence absolue désigne de manière absolue la ligne et la colonne de la cellule référencée, ceci se fait en faisant précéder les références aux ligne et colonne

du symbole $ (exemple $E$2) ; une référence relative, quant à elle, désigne de manière relative la ligne et la colonne de la cellule référencée (exemple E2).
La référence mixte est un mélange des deux modes de référence vus précédemment. La référence mixte permet de désigner la ligne de manière absolue et la colonne de manière relative (ou l'inverse) : ainsi écrire =$C4 dans la cellule D5 référence la cellule située dans la colonne C (référence absolue pour la colonne par l'utilisation d'un signe $) et dans la ligne précédente (référence relative pour la ligne, pas de signe $ devant le numéro de ligne).

2.5.4.- Bilan

Lorsque des calculs similaires ont lieu en plusieurs endroits du tableau, il faut se poser la question de l'utilisation des références absolues, relatives ou mixtes.
• Si je copie la formule sur une même ligne : est-ce que la cellule référencée doit se déplacer de la même façon ? Si oui, la référence sur la colonne doit être relative. Si non, la référence sur la colonne doit être absolue.
• Si je copie la formule sur une même colonne : est-ce que la cellule référencée doit se déplacer de la même façon ? Si oui, la référence sur la ligne doit être relative. Si non, la référence sur la ligne doit être absolue.

2.5.5.- Référence par nom

Au lieu de désigner une cellule par des coordonnées, on peut utiliser un nom, à condition que l'on ait défini auparavant à quelle cellule se rapporte ce nom. Dans le second exemple, on aurait pu référencer la cellule E2 par le nom MULTIPLICATEUR. La formule à écrire dans la cellule B5 aurait alors été =MULTIPLICATEUR*B4.

Pour pouvoir utiliser une référence par nom, deux étapes sont nécessaires.
1. Il faut d'abord affecter un nom à une cellule. Cela se fait en tapant le nom dans la fenêtre d'édition des noms ou par l'intermédiaire du menu Insertion/Nom/Définir.
2. On peut ensuite utiliser ce nom pour référencer cette cellule dans une formule. La fenêtre d'édition des noms permet d'introduire facilement des noms dans les formules.

L'utilisation de la référence par nom procure deux avantages.
1. Les formules deviennent plus lisibles : une formule du type
=MONTANT_HT*(1+TVA) est bien plus explicite qu'une formule du type
=$C$2*(1+$D$2) (en supposant que le nom de la cellule C2 est MONTANT_HT et celui de la cellule D2 est TVA) ;
2. En utilisant une référence par nom, la référence absolue de la cellule devient transparente.

Si pour une raison ou pour une autre on est amené à indiquer la TVA dans la cellule B2 (au lieu de la cellule D2), il suffit de faire porter le nom TVA sur cette nouvelle cellule. Ainsi les cellules utilisant la référence par nom ne seront pas à modifier au

contraire des cellules qui utilisaient la référence absolue $D$2, qu'il faudrait aller modifier à la main.

2.5.6.- Référence absolue et relative d'une plage de cellules

On peut avoir besoin de référencer plusieurs cellules adjacentes pour pouvoir effectuer certains calculs : pensons par exemple au calcul des notes moyennes d'un élève dans différentes matières. Si le calcul de la moyenne en français se fait sur les notes contenues dans les cellules B3, B4 et B5, une première solution serait d'écrire en B7 la formule =MOYENNE(B3;B4;B5) : on énumère toutes les cellules en séparant chaque référence par le signe << ; >>. Cependant, puisque les cellules référencées sont adjacentes, on peut écrire plus simplement =MOYENNE(B3:B5). Dans le cas général, pour désigner un ensemble de cellules adjacentes les unes aux autres et formant une plage rectangulaire, on se contente de désigner les deux cellules se trouvant aux extrémités haut-gauche et bas-droite de cette plage, en séparant toutefois la référence à chacune de ces cellules par le signe << : >>. En mode édition de formule, la désignation d'une plage peut être faite à la souris en sélectionnant la plage.

Si l'on a besoin de référencer des cellules se trouvant dans plusieurs plages (par exemple ici pour calculer la moyenne générale), il suffit de désigner chaque plage comme précédemment et de séparer les références à chaque plage par le signe << ; >>.
Ainsi la formule =MOYENNE(B3:B5;D3:D5) signifie : calculer la moyenne des valeurs se trouvant dans la plage délimitée par les cellules B3 et B5 (utilisation de << : >> entre B3 et B5) et (utilisation de << ; >>) dans la plage délimitée par les cellules D3 et D5 (utilisation de << : >> entre D3 et D5). Avec la souris, en mode édition de formule, il suffit de sélectionner les différentes plages tout en appuyant sur la touche Ctrl.

Noms et plages de cellules
Il est possible de nommer des plages de cellules de la même façon que pour les cellules.

Portée des noms
Les noms définis peuvent être utilisés dans tout le classeur ce qui signifie qu'un nom est défini pour tout le classeur. Ceci implique qu'il ne peut exister qu'une seule cellule ou plage de cellules associée à un nom. Par exemple, le nom taux_de_tva désignera une cellule unique d'une feuille du classeur.

2.5.7.- Organisation et choix du type de référence à utiliser

Comment choisir tel ou tel mode de référence ? Le choix d'un certain type de référence ne doit pas se faire au hasard, mais doit résulter d'un minimum de réflexion afin de construire efficacement sa feuille de calcul. Il faut avoir en tête les principes suivants.

• Les cellules isolées qui contiennent des données fixées doivent être référencées par nom, ce nom sera choisi pour décrire le contenu de la cellule (tva, tauxChange, réduction, ...).
• Pour garantir la cohérence de la feuille de calcul, il faut que chaque donnée n'apparaisse qu'une seule fois. Par exemple, il est hors de question que le montant de la TVA apparaisse dans deux cellules D2 et F6. Envisagez par exemple le jour où la TVA passe de 18,6% à 20,6% : que se passera-t-il si vous modifiez la cellule D2 et pas la cellule F6 ? La feuille de calcul risque fort de comporter des résultats incohérents !).
• Attribuer des noms aux plages de cellules à chaque fois que cela est nécessaire et possible. Les formules sont plus faciles à élaborer, à relire ou à corriger. Les erreurs de références sont moins courantes avec l'utilisation de noms.

2.6.- Formules et fonctions

Nous avons vu dans les deux paragraphes précédents comment faire référence à des cellules ou à des plages de cellules. Dans ce paragraphe, nous décrivons les formules qui peuvent être écrites dans un tableur. Rappelons que le tableur sait que le contenu d'une cellule est une formule si son contenu commence par le symbole =.

2.6.1.- Les expressions arithmétiques

Les expressions arithmétiques sont des expressions construites à l'aide des opérateurs arithmétiques usuels et de constantes, de références, de fonctions. Les opérateurs disponibles sont :

+ addition
- (unaire) prendre l'opposé
- (binaire) soustraction
/ division
* multiplication
^ élévation à la puissance.

Les règles de priorité usuelles sont respectées. Les parenthèses permettent de modifier les priorités. Les expressions arithmétiques portent sur des valeurs numériques. Le tableur convertit des valeurs. Il est du ressort du programmeur de vérifier la correction des expressions. Des exemples d'expressions arithmétiques sont :

=(B5+$B$2)*5,
=ventes-charges,
=somme(ventes)/5.

Remarque : pour le tableur, une date est une valeur numérique. Seul, le format d'affichage permet de visualiser cette valeur comme une date. Ceci permet d'effectuer des opérations arithmétiques. La plus utile est la soustraction. Par exemple, si la cellule de nom début contient la valeur correspondant au 5 mars 98, la cellule de

nom fin contient la valeur correspondant au 18 mars 98, la cellule de nom durée qui contient la formule fin-début aura pour valeur 13.

2.6.2.- Les expressions conditionnelles

Elles sont construites à l'aide d'opérateurs de comparaison. Ces expressions ont pour résultat une valeur logique VRAI ou FAUX. Les opérateurs de comparaison sont : = : égal à, > : supérieur strictement à, >= : supérieur ou égal à, < : inférieur strictement à, <= : inférieur ou égal à, <> : différent de. Par exemple, l'expression E2 <= 10 aura la valeur VRAI si la valeur contenue dans la cellule E2 est inférieure ou égale à 10 et la valeur FAUX sinon. On peut combiner des expressions conditionnelles pour construire des expressions logiques en utilisant les fonctions logiques (ET, OU, NON). Ceci est étudié dans le paragraphe sur les fonctions logiques.

2.6.3.- Les expressions de texte

Elles sont construites à l'aide de l'opérateur & qui permet de concaténer (mettre bout à bout) deux chaînes de caractères. Si la cellule de nom nom contient le texte << Digra >> et la cellule de nom prénom contient le texte << Omar >>, la cellule contenant la formule =prénom & nom aura pour valeur le texte << OmarDigra >>, la cellule contenant la formule =prénom & " " & nom aura pour valeur le texte << Omar Digra >>.

2.6.4.- Les fonctions

Les fonctions sont des outils de calcul puissants. Un tableur, Excel en particulier, met à votre disposition un large éventail de fonctions. Seul un utilisateur professionnel connaît toutes les fonctions et ses particularités. Vous utiliserez avec profit l'assistant fonction qui peut être appelé par un bouton dans la barre d'outils quand vous êtes en mode édition de formules, ou par l'appel du menu Insertion/fonction.

Nous ne présentons dans ce paragraphe que les fonctions usuelles. Il faut savoir utiliser l'aide (accessible facilement à partir de l'assistant fonction) pour rechercher une fonction et connaître son utilisation : sa syntaxe (comment l'écrire), ses paramètres (à quoi peut-on l'appliquer), son effet, ses restrictions.

Exemples de fonctions de date et d'heure

AUJOURDHUI

Cette fonction renvoie la valeur numérique correspondant à la date du jour ce qui permet d'obtenir la date du jour en utilisant le format Date. Cette fonction est une fonction sans paramètre (elle n'a pas d'argument). La syntaxe est AUJOURDHUI().

MOIS

Cette fonction renvoie le numéro du mois qui correspond à la date associée à la valeur numérique passée en argument. Cette fonction est donc une fonction à un paramètre. La syntaxe est MOIS(valeurnumérique). Par exemple, si la cellule de nom

datecommande contient la valeur numérique correspondant au 15/02/98, alors MOIS(datecommande) aura la valeur 2.

Exemples de fonctions de texte

MAJUSCULE
Cette fonction renvoie le texte passé en argument en majuscules. La syntaxe est MAJUSCULE(valeurtexte).
Par exemple, MAJUSCULE("bonjour") renvoie << BONJOUR >>. Si la cellule de nom nomclient contient la valeur << Dupond >>, MAJUSCULE(nomclient) renvoie << DUPOND >>.

CNUM
Cette fonction convertit le texte passé en argument en la valeur numérique correspondante. Si le texte ne peut être converti, la valeur #valeur! est renvoyée. Par exemple, CNUM("1200 F") renvoie 1200, CNUM("toto") renvoie une erreur.

Exemples de fonctions logiques

Les fonctions logiques permettent de construire des expressions logiques à partir d'expressions conditionnelles. Ces fonctions sont importantes et nous en donnons ici la liste complète.

ET
Cette fonction renvoie la valeur VRAI si tous ses arguments ont la valeur VRAI, et FAUX sinon. Cette fonction admet un nombre d'arguments compris entre 1 et 30. La syntaxe est

ET(valeurlogique1;valeurlogique2;...).

Par exemple, ET(A1<4;A1>=0) vaut VRAI si A1 contient la valeur numérique 3, vaut FAUX si A1 contient -1 ou 7.

OU
Cette fonction renvoie la valeur VRAI si l'un au moins des arguments a la valeur VRAI et FAUX sinon (c'est-à-dire si tous les arguments ont la valeur FAUX). Cette fonction admet un nombre d'arguments compris entre 1 et 30. La syntaxe est

OU(valeurlogique1;valeurlogique2;...).

Par exemple, OU(A1<7;A1>=10) vaut VRAI si A1 contient la valeur numérique 3 ou 15, vaut FAUX si A1 contient 8.

NON
Cette fonction renvoie la valeur logique contraire de la valeur logique passée en argument. La syntaxe est

NON(valeurlogique).

SI

Cette fonction renvoie une valeur ou une autre selon la valeur de vérité d'une expression logique. La syntaxe est

SI(testlogique;valeursiVRAI;valeursiFAUX).

Le premier argument testlogique doit être une expression logique de résultat VRAI ou FAUX, le deuxième argument est la valeur retournée par la fonction si l'expression logique a la valeur VRAI, le troisième argument est la valeur retournée par la fonction si l'expression logique a la valeur FAUX. Par exemple, si la cellule de nom ventes contient la valeur 1200, si la cellule de nom charges contient la valeur 1500 (respectivement 1000), la formule

SI(ventes>=charges;"excédent";"déficit")

renvoie le texte << déficit >> (respectivement << excédent >>).

Le test logique peut être une expression logique, par exemple :

SI(ET(moyenne>=12;moyenne<14);"assez bien";"autre").

On peut imbriquer les si, par exemple :

SI(moyenne >=12;"a.bien"; SI(moyenne >=10;"honorable";"échec")).

Exemples de fonctions mathématiques

Toutes les fonctions mathématiques et trigonométriques usuelles sont disponibles (SIN, LN, RACINE,...).

ARRONDI

Cette fonction renvoie la valeur numérique donnée comme premier argument arrondie au nombre de chiffres passé en second argument. La syntaxe est ARRONDI(valeurnumérique;nombrechiffres). Par exemple, ARRONDI(29,372;2) vaut 29,37 ; ARRONDI(29,372;1) vaut 29,4 ;
ARRONDI(29,372;0) vaut 29 ; ARRONDI(29,372;-1) vaut 30.

SOMME

Cette fonction renvoie la somme de toutes les valeurs numériques passées en argument. La syntaxe est : SOMME(argument1;argument2;...). Les arguments peuvent être des valeurs numériques, mais le plus souvent les arguments seront des références à des plages de cellules. Dans ce cas, seules les valeurs numériques sont prises en compte dans le calcul de la somme.

Exemples de fonctions statistiques

La plupart des fonctions statistiques sont disponibles (médiane, écart-type, variance, ...). Nous ne détaillons dans ce paragraphe que les fonctions les plus basiques. En général, les arguments seront des références à des plages de cellules comme pour la fonction SOMME.

MAX
Cette fonction renvoie la plus grande valeur numérique de la liste des arguments. La syntaxe est : MAX(argument1;argument2;...).

MIN
Comme MAX mais MIN !

MOYENNE
Cette fonction renvoie la moyenne arithmétique des valeurs numériques de la liste sans prendre en compte les autres valeurs. La syntaxe est :
MOYENNE(argument1;argument2;...).

NB
Cette fonction renvoie le nombre de valeurs numériques dans la liste des arguments. La syntaxe est : NB(argument1;argument2;...).

NBVAL
Cette fonction renvoie le nombre de cellules non vides (contenant des valeurs numériques ou pas) dans la liste des arguments. La syntaxe est : NBVAL(argument1;argument2;...).

3.- LISTES ET TABLEAUX CROISES

Lorsque l'on manipule des quantités de données importantes telles que des listes de clients, des listes de produits, des résultats de ventes, il est important de pouvoir synthétiser les données pour servir de base aux décisions. Dans ce chapitre, nous étudions les fonctionnalités fournies par un tableur pour ce type de traitements.
Une liste est constituée d'une suite d'enregistrements. Tous les enregistrements ont la même structure. La structure d'une liste est définie par un certain nombre de champs. Chaque champ porte un nom qui correspond à l'en-tête de colonne. Les champs ont une valeur toujours prise dans un type, c'est-à-dire, l'ensemble de ses valeurs possibles.
Il existe deux modes de représentation pour les listes : le mode tableau et le mode grille. On passe d'un mode de représentation à l'autre en utilisant le menu Données-Grille. Le mode tableau est le mode usuel, le mode grille présente à l'utilisateur une fiche par enregistrement avec la liste des valeurs de ses champs.

3.1.- Opérations sur des enregistrements

Les opérations permises sont souvent indépendantes du mode de représentation choisi.

3.1.1.- Parcourir/Rechercher

En mode tableau, on parcourt la liste à l'aide des touches de déplacement ou des ascenseurs ; pour rechercher, on peut utiliser le menu Édition-Rechercher. En mode grille, on parcourt la liste à l'aide des boutons Précédente et Suivante ou des ascenseurs ; pour rechercher, on précise les critères de recherche après l'appui sur le bouton Critères.

3.1.2.- Ajouter/Supprimer/Modifier

Dans le mode tableau, ces actions sont réalisées par des actions sur les lignes ou les cellules. Pour ajouter un enregistrement, il suffit d'insérer une ligne, puis de renseigner les cellules. Il est conseillé d'insérer plutôt que d'ajouter une nouvelle ligne. En effet, lors de l'insertion d'une ligne, la taille de la liste est modifiée, et si vous avez nommé votre liste, le nom portera maintenant sur la liste avec le nouvel enregistrement compris. Supprimer un enregistrement consiste à supprimer une ligne. Modifier un enregistrement consiste à modifier les contenus des cellules de la ligne. Dans le mode grille, ces opérations sont facilitées à l'aide de boutons : Nouvelle pour ajouter, Supprimer pour supprimer, les modifications sont faites dans la grille, on peut annuler une modification à l'aide du bouton Rétablir.

3.1.3.- Trier

On peut classer les enregistrements dans différents ordres. Un ordre est défini en précisant des critères de tri. Par exemple, la liste des personnes peut être triée dans l'ordre alphabétique de noms ou de nombres. Pour appliquer un tri, il suffit de se placer n'importe où dans la liste et d'utiliser le menu Données-Trier.

Spécifications : vous devez préciser la zone de liste par son nom ou par les références à la plage de cellules et préciser les critères de tri comme dans les exemples précédents.

3.1.4.- Filtrer/Extraire

Filtrer consiste à ne faire apparaître dans la liste que les enregistrements vérifiant certains critères. Extraire consiste à recopier la sous-liste des enregistrements vérifiant certains critères à un autre endroit dans le classeur. On peut filtrer à l'aide des filtres automatiques ou des filtres élaborés, on peut extraire à l'aide des filtres élaborés.

Filtres automatiques

On utilise le menu Données-Filtrer et le sous-menu filtre automatique. On peut alors filtrer sur les différents champs. Lorsqu'on filtre sur plusieurs colonnes, on filtre selon la conjonction des critères.

Filtres élaborés
Les filtres élaborés permettent de faire des filtres qui ne peuvent être réalisés à l'aide de l'outil Filtre automatique. Ce sont les filtres pour lesquels les critères de filtre contiennent des disjonctions (des ou) ou contiennent des expressions calculées. Ils permettent également de réaliser des extractions. Il faut préciser la zone source (le nom ou les références de la plage contenant la liste), la zone de critères (le nom ou les références de la plage contenant les critères de filtre), la zone de destination (la cellule à partir de laquelle on va ranger la liste). Dans la définition des critères, on utilise la convention suivante : les critères sur une même ligne correspondent à une conjonction, les critères sur une ligne différente à une disjonction.

Sous-totaux
Les sous-totaux permettent d'ajouter à la liste des lignes récapitulatrices, par exemple un décompte, une moyenne, une somme. Les sous-totaux portent sur des groupes. Un groupe est formé par des suites d'enregistrements pour lesquels la valeur d'un champ est identique.

Avertissement : pour que les groupes soient correctement formés, il faut préalablement trier sur le champ sur lequel doit porter le groupe. Pour faire des sous-totaux, on utilise le menu Données-Sous-totaux. Les fonctions de synthèse les plus courantes sont : Moyenne, Somme, Nb, Nbval, Min, Max.

3.2.- Tableaux croisés dynamiques

Qu'est-ce que c'est ?
Les tableaux croisés permettent de synthétiser des données en regroupant des résultats sous forme de tableaux en fonction de différents critères. Une entreprise effectuera par exemple des études de ventes de produit par produit, par gamme de produit, par gamme de produit par région... Les tableaux croisés Excel permettent de synthétiser des données provenant de une ou plusieurs listes. On peut facilement retirer ou ajouter des critères, ils peuvent être réactualisés lorsque les données d'origine sont modifiées. Les exemples présentés dans cette section utilisent la liste des élèves de la section précédente. Pour créer un tableau croisé, il faut utiliser le menu Données-Trier et vous devez préciser :
la zone source : la zone de liste par son nom ou par les références à la plage de cellules,
la zone destination : la feuille dans laquelle sera placé le tableau résultat,
les champs de page,
les champs de ligne,
les champs de colonne,
les champs valeurs et les opérations de synthèse associées.

Que peut on faire ensuite ?
La barre d'outils des TCD (Tableaux Croisés Dynamiques), montre les actions possibles sur un TCD construit. Pour réaliser ces actions il est d'abord nécessaire de se placer dans un TCD construit.

Actualiser
Si les données d'origine, c'est-à-dire les données contenues dans la liste, ont changé, le TCD peut être recalculé avec cette action.

Assistant
On lance l'assistant pour permettre de modifier le TCD.

Champ dynamique
Si la sélection est sur
- une valeur (au milieu du tableau), on peut modifier l'opération de synthèse ; on peut ajouter ou supprimer un champ valeur ;
- un élément ou un champ de ligne ou de colonne, on peut modifier les sous-totaux ou masquer des éléments.

Grouper ou dissocier
Cela concerne les éléments en ligne, en colonne ou en page. C'est très intéressant surtout s'il y en beaucoup. Exemple : des dates, des notes, des sommes. On peut grouper de deux façons :
automatique : on donne une base et un incrément (exemple : le CA est présenté à partir de 0 puis de kilofrancs en kilofrancs) ou, dans le cas de dates ou d'heures, on regroupe par minute, heure...mois, trimestre, année, ...
par sélection : on sélectionne à la souris les champs que l'on veut regrouper.
en affichant des détails : Excel construit une nouvelle feuille de calcul montrant tous les élements qui ont contribué au résultat. On sélectionne un élément en ligne ou colonne, on peut afficher pour cet élément des éléments d'autres champs.

4.- PRESENTATION

Le tableur peut aussi aider à présenter et analyser des données. Il permet par exemple de construire des graphiques et de mettre en page ses résultats sous forme lisible et agréable. La couleur, les encadrements et les dessins permettent de mettre en valeur certains éléments d'un tableau, d'afficher des histogrammes et des camemberts. Le document ainsi constitué peut ensuite être imprimé ou diffusé sur un réseau.

C H A P I T R E VII

L'Internet

NOTE LIMINAIRE

Ce texte est extrait d'un document WWW écrit par Gille Maire qui était disponible sur Internet (http://www.imaginet.fr/ime/reseaux.htm). La version reprise ci-dessous s'inspire des pages WWW précitées.

1.- QU'EST-CE QU'UN RÉSEAU

Le mot réseau vous fait-il peur? Il n'y a pas de quoi. Vous en utilisez un tous les jours, c'est le réseau téléphonique. Mais ce réseau, nous le verrons, n'a pas que des avantages, du moins quand il est utilisé avec son terminal le plus commun, le téléphone. Sur le réseau téléphonique, vous connaissez le téléphone, le fax, le Minitel, vous connaissez maintenant Internet.

La première chose qui surprend lorsqu'on entre dans le monde magique d'Internet est sa soudaine proximité avec le Japon et New York. Vous pouvez communiquer avec la même facilité entre votre domicile et New York ou entre votre domicile et votre voisin de palier. Cette figure de style est volontaire : dans ce monde où on ne connaît plus son voisin de palier, il n'est pas dit qu'Internet vous aidera à communiquer plus facilement qu'aujourd'hui avec vos contemporains!

A la différence du réseau téléphonique, qui techniquement ne vous empêche pas de communiquer avec les quatre coins du monde, la communication Internet est gratuite. Lorsque vous avez un accès Internet, vous payez éventuellement le prix de la communication entre votre domicile et votre fournisseur Internet, mais pas de supplément de communication que vous communiquiez avec le Japon ou la Suisse. On sait donc que la communication du particulier avec un autre particulier ou avec une entreprise passera de plus en plus par Internet. Internet vous offre même la communication avec des gens regroupés autour de vos centres d'intérêt dans des forums ou des canaux.

Si on ne parle plus de communication, mais d'accès à des volumes d'information, on peut aussi dire que le jour où vous vous connectez sur Internet, vous disposez dans votre bureau de la plus formidable bibliothèque de documents que l'humanité n'ait jamais fournie à personne. Ce n'est pas pour autant que vous lirez tous les livres. Ce n'est pas pour autant que vous aurez envie de lire!

2.- LES ADRESSES SUR LE RÉSEAU INTERNET

Votre bibliothèque, même si elle est en désordre, vous permet de retrouver sans trop de problèmes tous les livres que vous aimez. Par contre, lorsque vous vous rendez dans votre bibliothèque municipale, il vous est, la plupart du temps, nécessaire de consulter un fichier pour retrouver la trace du livre que vous aimeriez lire. Il en va de même avec les adresses de vos amis, vous en connaissez peut-être par coeur, mais vous avez un répertoire pour noter les adresses des moins proches. Heureusement pour ceux que vous avez un peu perdus de vue, il y a toujours les pages de l'annuaire ou du Minitel.

On trouve trois types d'adresse pour relier un service Internet ou une personne :

- son adresse Internet,
- son numéro IP,
- son adresse URL.

Les trois méthodes sont équivalentes, la troisième est la plus en vogue aujourd'hui et nous verrons pourquoi.

2.1.- L'adresse Internet

Sur Internet le nom d'un site centralisant plusieurs personnes est organisé en domaines. En jargon Internet, les noms sous la forme

`nom@organisation.domaine`

sont dits FQDN (Fully Qualified Domain Name).

Voilà, plus de mystère, `chirac@elysee.fr` est un français (fr), il travaille à l'Elysée et son nom est Chirac. On peut noter qu'un nom peut être composé, par exemple `Gilles.Maire@xerox.fr` désigne un Français qui travaille chez Xerox et dont le prénom est Gilles. Par exemple, paul.prevost@sci.ups.edu travaille dans l'éducation, peut-être dans une université nommée ups, dans le département sci.

Les noms de domaines sont regroupés en grandes classes :

com	désigne les entreprises commerciales,
edu	désigne l'éducation,
gov	désigne les organismes gouvernementaux,
mil	désigne les organisations militaires,
net	désigne les organismes fournisseurs d'Internet,
org	désigne les autres organismes non référencés.

Ils sont aussi regroupés en pays :

be	Belgique,
ca	Canada,
fr	France,
uk	United Kingdom,
etc…	

Il est important de considérer que la langue anglaise est la langue d'Internet mondial, par contre dès que vous communiquez avec une adresse se finissant par l'or-

ganisation .fr, le français est de rigueur. Ceci n'est pas vrai pour les Belges (.be), les Canadiens (.ca) qui parlent plusieurs langues.

2.2.- Les numéros IP

Il existe un équivalent des noms précédemment définis, c'est un numéro de 32 bits, que l'on écrit par quatre nombres séparés par trois points.
Par exemple, 192.203.245.63 est une adresse TCP/IP donnée sous une forme plus technique mais moins mnémotechnique que la précédente. Ce sont ces adresses que connaissent les ordinateurs qui communiquent entre eux. Là aussi on retrouve une certaine logique d'attribution de ces numéros. Le premier groupe de numéros peut être plus ou moins grand (on dit de classe A, B ou C), de telle sorte que plus on réserve de digits pour les premiers numéros, moins il en reste pour la deuxième partie de l'adresse.
Ces numéros IP ne peuvent pas être donnés arbitrairement puisque deux ordinateurs sur l'Internet ne peuvent pas avoir le même numéro. C'est un organisme appelé NIC (Network Information Center) qui fournit les premiers numéros appelés racine du numéro IP. Charge à l'administrateur de votre réseau de vous distribuer les numéros disponibles dans la plage de numéros attribués.

2.3.- Les adresses URL (Unified Resource Locator)

Avec les dernières technologies, la tendance est de donner les adresses directement sous la forme d'hypertexte ou d'URL. Comment cela marche-t-il?
D'abord, on place le type de service auquel on s'adresse. Nous verrons que les services peuvent être des serveurs de Web (http), des serveurs de fichiers (ftp), et d'autres que nous expliciterons plus en détail dans le chapitre sur les Web.
Donc une adresse URL est une adresse de la forme :

service://machine/repertoire/fichier

par exemple :

http://www.microsoft.com
http://www.ulg.ac.be

L'avantage de ce type d'adresse est qu'il englobe beaucoup plus d'informations que l'adresse FQDN, puisqu'il comprend :

- l'adresse FQDN,
- le type de service,
- l'emplacement sur le serveur,
- le nom du fichier.

Souvent le nom du répertoire d'accueil est omis ainsi que le nom du fichier, car le nom service://machine est non ambigu.
Notons que par défaut votre lecteur de Web acceptera même des adresses URL sans les symboles http://. Ainsi l'adresse www.microsoft.com est suffisante pour se rendre chez Microsoft.

2.4.- Les serveurs de noms

Vous devez juste vous souvenir que généralement vous pouvez (en tant qu'utilisateur) utiliser les adresses FQDN ou les numéros IP indifféremment. Par contre les ordinateurs, entre eux, ne connaissent que les numéros IP.
Concrètement, lorsque vous donnez une adresse FQDN à un ordinateur, celui-ci interroge un serveur, qu'il contacte automatiquement, pour la transformer en numéro IP.
Chaque domaine est servi par un ordinateur, que l'on appelle serveur de noms, qui est chargé de faire cette transformation d'adresse en numéro IP.

3.- LES CONNEXIONS PHYSIQUES

Mais comment sont connectés les ordinateurs entre eux pour faire ce très gros réseau ?

3.1.- Les liaisons téléphoniques

Les liaisons téléphoniques vont jusqu'à 56k bits par seconde. Avec de telles liaisons, on peut transporter un fichier de 1 Mo en 3 à 4 minutes théoriques, c'est-à-dire en fait en 8 à 10 minutes. Ce qui veut dire que pour transférer un CD audio (640 Mo) il faudra 4 jours. Les lignes téléphoniques sont des lignes qui laissent passer la voix, (ce qu'on appelle un signal analogique, par opposition au signal numérique). Pour transformer le signal numérique en un signal analogique compatible avec une ligne téléphonique, on utilise un Modulateur-Démodulateur, appelé modem. Si vous écoutez le bruit que fait votre modem lorsque son haut parleur est allumé, vous entendez le signal analogique, c'est-à-dire le son produit par le signal numérique.

3.2.- Les liaisons numériques

Les liaisons numériques laissent passer le signal numérique, point n'est besoin de transformation de signal numérique en signal analogique, et il ne faut plus parler de modem numérique mais d'adaptateur. Cette erreur, souvent commise, est due au fait que les modems externes ressemblent aux adaptateurs numériques. Lorsque vous vous connectez chez un fournissseur d'accès par une liaison numérique, la numérotation est immédiate.

3.3.- Les liaisons spécialisées et l'ADSL

Les liaisons spécialisées (LS) vont de 1 à 10 Mbps pour les plus courantes. Cela revient à transférer un CD audio (640 Mo) en une trentaine de minutes. Aujourd'hui les lignes à 1,5 Mbps sont classées sous l'appellation T1 et les lignes à 45 Mbps sous l'appellation T3.
Le standard actuel (depuis 1995) pour les connexions privées est l'ADSL (Asymetric Digital Suscriber Line - Ligne à capacité asymétrique). L'ADSL est un protocole utilisé dans le transfert de données sur ligne téléphonique fixe où la capacité de débit descendante (ie : DOWNLOAD - Serveur Internet vers PC utilisateur) est supérieur

(souvent d'un facteur ≈10) à la capacité montante (ie : UPLOAD - PC utilisateur vers Serveur internet). En pratique un signal à haute fréquence est envoyé sur le fil et ce signal est récupéré par un adaptateur spécial qui le décode. L'ADSL s'auto-adapte à la qualité du signal reçu qui dépend de la longueur et de la qualité du câblage. La vitesse de la connexion varie donc avec les conditions de transmission. En pratique, pour disposer de l'ADSL, il faut être situé à moins de 5 à 7 km de la dernière borne de retransmission. Les fréquences des porteuses utilisées par l'lADSL (souvent entre 100 à 300 kHz - max. 1.1MHz) sont largement supérieures aux fréquences vocales et n'interfèrent pas avec l'utilisation classique du téléphone (qui fonctionne entre 1 et 20 kHz env.).
On peut connecter plusieurs ordinateurs sur une ligne ADSL. Pour ce faire, on connecte un réseau local Ethernet sur un routeur connecté sur la liaison extérieure. Il est possible de transmettre des émissions de télévision par cette technique.
Consulter : http://www.commentcamarche.net/technologies/adsl.php3

4.- HISTOIRE D'INTERNET

C'est par l'ARPA (U.S Defense Department's Advanced Research Projects Agency) que tout a commencé dans les années 1960 sur le réseau téléphonique avec la technologie des commutations de paquets, agrémentée d'une dose d'automatisation (pour que les paquets d'informations trouvent leur chemin pour aller d'un ordinateur à un autre... en passant par un réseau d'autres ordinateurs).
Le système devait être capable de s'auto-configurer si un ou plusieurs maillons venait à défaillir (par exemple en cas d'attaque nucléaire). Il ne devait donc pas avoir de structure centralisée de gestion du réseau et tout les noeuds devaient être capable de se reconfigurer dynamiquement. Le système fut nommé ARPANET (c'est-à-dire le réseau de l'ARPA). Le réseau initial ne disposait que du courrier électronique. Mis dans le domaine public, il fut repris par les universitaires qui y virent une occasion de faire des conférences au moyen du courrier électronique.
Puis, dans les années 1970, l'ARPA continua ses recherches dans l'étude des protocoles de transfert de données entre des réseaux d'ordinateurs, réseaux qui pouvaient être de natures différentes.

Le nom d'Internet (qui élargissait l'ARPANET à l'Inter networking) fut alors adopté et développé entre les différentes universités américaines. Dans les années 1980, le réseau Internet a commencé son expansion, non plus au travers de l'armée mais au travers des universités mondiales, des laboratoires de recherche, des grosses entreprises.
Un certain nombre d'entreprises de taille moyenne ou des particuliers passionnés ont décidé de s'unir pour créer des services privés, de là sont nés les fournisseurs privés de connexions Internet. Depuis les années 1990 ont a vu le Net continuer à grossir à une vitesse exponentielle, de 10 à 20 % par mois sous l'impulsion du Web.

5.- LE WEB (WORLD WIDE WEB)

5.1.- Introduction et définition

L'outil qui rendit populaire l'Internet c'est incontestablement le 3W, le WWW, le World Wide Web en un mot le Web. Le mot Web désigne en anglais la toile d'araignée et World Wide Web désigne donc la toile d'araignée couvrant le monde entier.
L'outil est graphique, il est puissant, et très facile à utiliser, il est beau et il ne coûte pas cher. Par le Web vous pouvez visiter une exposition, lire votre journal, apprendre l'anglais, commander une Pizza. De plus tous les connectés ont leur page Web personnelle.
D'un point de vue technique, le WWW relie des serveurs HTTP qui envoient des pages HTML à des postes dotés d'un navigateur. Le protocole de communication entre les navigateurs et les serveurs est basé sur le principe des hypertextes (Hyper Texte Transfert Protocol). Le langage permettant de décrire les pages Web est le HTML (Hyper Text Markup Langage).
Ce langage à balises permet de doter certains mots ou images d'un d'hyperlien. L'hyperlien est constitué d'une adresse que vous atteindrez en cliquant sur le mot ou l'image doté de l'hyperlien.
Voilà défini le World Wide Web : une toile d'araignée de serveurs d'informations reliés les uns aux autres par des liens physiques (le réseau matériel) et des liens logiques (les liens hypertextes). Ces liens hypertextes permettent de voyager d'un serveur à l'autre sur le réseau Internet.
Et le plus spectaculaire dans les Web est bien la large utilisation de ces liens hypertextes, ce sont des textes de couleur différente (bleu en général) ou des images sur lesquelles vous cliquez pour vous retrouver dans un autre document. Ces hypertextes rendent votre lecture plus dynamique : si vous lisez un article traitant de "l'invention du téléphone sans fil" et que votre article cite ATT, un hypertexte vous permettra de voir une présentation d'ATT en cliquant simplement sur le mot ATT. Des documents contiennent ainsi des références sur d'autres documents, créant une toile d'araignée de documents recouvrant le monde.
D'un point de vue théorique, les Web sont autant de points d'informations se recoupant par des liens et garantissant au NetSurfer (c'est à dire celui qui se promène de Web en Web) des informations toujours mises à jour. Il y a plus de 100 millions de pages de Web dans le monde à ce jour sans compter les Web personnels faits par les utilisateurs passionnés.
Les hypertextes peuvent adresser d'autres documents de type Web mais aussi des serveurs de fichiers, des serveurs de News etc…

Pour accéder à une adresse depuis votre lecteur de Web, vous avez trois possibilités :
• cliquer sur un hypertexte de couleur différente dans la page Web,
• passer par le menu File puis Open/URL et, dans la fenêtre qui apparaît, entrer l'adresse http:// correspondant au site auquel vous désirez accéder,

• si vous utilisez Netscape ou Internet Explorer, il est inutile de passer par le point ci-dessus si le bandeau Location en dessous de la barre de menu est présent : entrez directement le nom du site à visiter.

5.2.- Les URL

5.2.1.- Le concept d'URL

Tout document du Web est identifié par une référence univoque, son URL. Les URL (Uniform Resource Locators) sont les noms donnés aux hypertextes. Un URL peut être un serveur ftp, un fichier sur votre disque, un serveur gopher, une image, une adresse courrier, un serveur de News, un serveur telnet et bien sûr un serveur http, c'est-à-dire un serveur de Web.
Un exemple d'URL est http://www.xerox.fr:8080 un autre est ftp://inria.ftp.fr

En général, vous n'avez pas à connaître précisément les adresses de service puisque vous avez à cliquer sur un texte de couleur vous y conduisant. Ainsi, le Web numéro 1 vous conduit au Web numéro 2 qui vous conduit au Web numéro 3 et ainsi de suite.
Le point sensible des Web est que les adresses pointées par les URL ont tendance à changer souvent, soit parce que le propriétaire du Web a déménagé ou soit parce que le serveur est devenu saturé. Donc, il n'est pas exclu qu'en surfant vous tombiez sur quelques impasses!
Il se peut que vous vouliez aller un peu plus vite et atteindre directement une page HTML sans progresser de proche en proche. Ce sera possible en donnant l'adresse URL de votre cible. Ceci se fait dans la majorité des cas en cliquant sur le menu file et open URL de votre lecteur de Web.
Parfois vous aurez des références d'adresses de la forme :
http://serveur/directory/fichier.html
où fichier.html est un nom de fichier HTML sur le serveur de Web (html signifiant HyperText Markup Language). Ceci signifie notamment que le fichier n'est pas sur un PC car sur celui-ci les extensions de fichier n'ont que trois caractères.
En revanche :
http://serveur.directory.fichier.htm
est une adresse de fichier HTML qui a de grandes chances de se trouver sur un PC.
En général, pour trouver le Web d'une compagnie commerciale, par exemple Compaq, il suffit de taper l'adresse de cette société : http://www.compaq.com. Ceci se vérifie pour Microsoft, Xerox, IBM et bien d'autres.
La plupart du temps vous n'aurez pas à taper ces longues syntaxes car vous aurez les pages les plus fréquemment utilisées dans votre liste de pages favorites, ou bien vous aurez trouvé l'adresse en passant par un autre Web.

5.2.2.- Types d'URL

On trouve plusieurs types d'URL, chacun représentant un service donné. Il est important de garder présent à l'esprit que l'architecture WWW permet de couvrir l'en-

semble des services cités. Les noms d'URL utilisent les lettres de l'alphabet, en général en minuscule, sachant que les noms en majuscule et minuscule sont équivalents. Les chiffres sont autorisés. Certains caractères / . : # ont une signification particulière et enfin certains caractères sont dits non sûrs dans la mesure où ils sont interprétés : les blancs, les étoiles, etc.
Passons en revue les URL les plus usitées, mais auparavant regardons la forme générale d'un URL. Un URL est de la forme :

service://<user>:<password>@<host>:<port>/<url-path>

Mais nous ne donnerons pas l'identificateur de password (mot de passe) et de user (nom de login) pour ne pas compliquer l'exposé.

file

file:///repertoire/fichier.txt

Ce type d'URL permet d'accéder à un fichier, ici fichier.txt, présent sur votre disque. Ce type d'URL n'est pas très normalisé et on trouvera :

sur VAX /VMS : file://vms.host.edu/disk$user/my/notes/note12345.txt

sur PC : file:///cltmp/fichier.txt

http

http://serveur:port/repertoire/fichier.html

Ce type d'URL permet d'accéder à un serveur Web, généralement présent sur une autre machine. Le plus souvent ni le port, ni les noms de répertoires, ni le nom de fichier ne sont mentionnés. Ils ont des valeurs par défaut.
Note : l'URL d'un serveur http peut être suivi d'un point d'interrogation et d'une chaîne de caractères servant de chaîne de requête sur le serveur http.

ftp

ftp://serveur/repertoire/fichier

Ce type d'URL permet d'accéder à un serveur ftp et de visualiser l'ensemble des fichiers d'un répertoire si aucun fichier n'est spécifié, de rapatrier le fichier sur votre disque local si un nom de fichier est donné. Un service ftp vous permet d'écrire ou de lire des fichiers à distance sur une autre machine du réseau.

mailto

mailto:nom@organisation.domaine

Ce type d'URL permet d'écrire un courrier électronique à l'utilisateur dont l'adresse figure dans l'URL.
Certains navigateurs acceptent la syntaxe étendue :

mailto:nom@organisation.domaine?subject=SUJET

permettant de renseigner la zone sujet dans le courrier.

telnet

telnet://Nom:Password@serveur:port

Ce type d'URL permet d'ouvrir une session telnet. Une session telnet est une fenêtre représentant la console d'une machine distante, présente sur Internet.

gopher
gopher://serveur:port/repertoire/fichier#marqueur
Ce type d'URL permet d'ouvrir un menu Gopher. Un menu Gopher est une arborescence de fichiers plus animée que celle que l'on trouve sur ftp. C'est l'ancêtre du WWW (les documents sont exclusivement textuels).

news
news:nom.de.la.news ; news:numero de news
Ce type d'URL permet d'ouvrir une News sur le serveur de News paramétré dans votre logiciel. Les news sont des forums où des courriers restent stockés par thème pendant une durée de quelques jours.
Notes : lorsque l'URL est demandé avec le nom de la news, il est possible de mettre un astérisque (*), pour spécifier un ensemble de news, par exemple news:alt.binaries.* affichera toutes les News commençant par alt.binaries.

Autres
En principe, d'autres services peuvent être déclarés par les serveurs http et le lecteur de pages Web peut être configuré pour accéder à ces nouveaux services. En pratique, aucun serveur http n'utilise d'autres services que ceux décrits ci-dessus sauf pour expérimentation. Un service expérimental doit avoir un nom commençant par x-

5.2.3.- Les logiciels de lecture de Web

Les logiciels de lecture de Web sont appelés des fureteurs, des navigateurs, des butineurs ou des "browsers". Le terme préféré est aujourd'hui navigateur.
Le premier logiciel lecteur de Web graphique connu par le public fut Mosaic de NCSA (National Center for Supercomputing Applications) de l'université de l'Illinois. Il fut précédé de logiciels précurseurs comme tkWWW, violaWWW et midaswww dans l'année 1992. Ces logiciels, s'ils possédaient des options très en avance sur ceux d'aujourd'hui, n'ont pas connu le succès du logiciel du NCSA.
Les logiciels Internet Explorer, Firefox et Safari couvrent aujourd'hui presque tout le marché. Ces navigateurs proposent des possibilités d'ajout de composantes logicielles appelées Plug-Ins, capables d'effectuer des tâches évoluées. Les navigateurs reconnaissent aujourd'hui les langages JAVA, VRML.
Les logiciels lecteurs de Web modernes sont polyvalants, ils permettent de charger des fichiers par ftp, de lire des News, d'envoyer une session telnet et ont tendance à remplacer les multiples programmes utilisés hier.

5.2.4.- Les chercheurs d'informations

Il existe des Web spécialisés dans la recherche d'information sur l'Internet. Ces Web sont couplés avec des bases de données qui sont alimentées en permanence. Ils

permettent de retrouver n'importe quel type d'information, que ces informations soient stockées sur un Web ou sur un serveur ftp. En pratique, ils sont presque exhaustifs dans le monde des Web, et encore assez pauvres dans le monde des serveurs ftp. Il est important que vous mettiez un de ces chercheurs dans votre liste de pages favorites après les avoir utilisés et choisi celui qui vous convenait le mieux.

Principe de fonctionnement

D'un point de vue théorique, les moteurs de recherche, au sens véritable du terme, sont ceux qui effectuent eux-mêmes la recherche et l'indexation des pages Web sans intervention humaine. Les sites d'indexation automatique comprennent tous :

- une base de données,
- un logiciel de mise à jour de cette base de données.

Nous allons expliciter brièvement le fonctionnement d'un programme de mise à jour en sachant que ceci mériterait un chapitre à part entière. Ces logiciels de mise à jour, sont appelés Robots, nom qui indique bien qu'ils correspondent à des programmes automatiques. Un robot est un programme simple dans le principe, mais que les optimisations rendent complexes dans leur programmation.

Tout d'abord, ces programmes ont deux missions essentielles :

- lire l'information et la gérer,
- chercher dans ces informations d'autres adresses où aller chercher.

La première de ces deux étapes est facilement compréhensible. Elle permet de faire de l'indexation textuelle qui revient à mémoriser des mots clés, éventuellement les phrases dans lesquelles ils apparaissent et surtout leur localisation, c'est-à-dire leur adresse URL. La recherche des mots clés se fait par des logiciels comme Glimse, agrep ou free Wais qui sont plus ou moins performants et qui ont des fonctionnalités plus ou moins évoluées.

Ces logiciels lisent donc un fichier et mettent dans un index les mots lus dans le fichier. Dans cette lecture, ils analysent parmi les mots rencontrés, les adresses URL, de façon à connaître de nouvelles adresses de Web à explorer par la suite. Cette analyse doit être assez fine pour prendre en compte les aspects suivants :

- mémoriser l'adresse IP du site trouvé et les noms des fichiers correspondants, pour éviter la redondance et le bouclage des noms de serveur,
- ne pas appeler les programmes CGI, les pages ISINDEX, afin de ne pas provoquer des requêtes intempestives (de même, les URL mailto: ou telnet: ne sont pas exécutés),
- mémoriser les dates de visite des pages de manière à ne plus repasser pendant une durée paramétrée.

C'est ainsi que les programmes de recherche scrutent en permanence Internet. Et lorsque vous demandez à GOOGLE ou YAHOO de chercher un mot clé, il effectue la recherche non pas sur Internet mais dans son fichier de recherche.

5.2.5.- Histoire du Web

Le Web est un protocole nouveau qui est basé sur des concepts assez anciens. Regardons de plus près cette histoire du Web qui débuta avant le nom et qui atteint sa maturité à l'aube du 21ème siècle.

1945 : Vannevar Bush, conseiller de Roosevelt, publie une note concernant des toiles conceptuelles d'information.

1965 : Ted Nelson donne naissance à l'Hypertexte, puis à un logiciel de navigation hypertexte qui ne fonctionnera jamais.

1987 : Hypercard, logiciel d'Apple utilisant les Hypertextes, est lancé.

Mars 1989 : Tim Berners-Lee du CERN publie l'article " Hypertexte et le CERN ".

Octobre 1991 : Le premier Web fonctionne au CERN en mode texte et sous NExT Step avec le premier navigateur intitulé World Wide Web. Cette première version de navigateur sur ce système d'exploitation confidentiel mais au combien en avance sur son temps, comprenait également une partie éditeur HTML Wysiwyg.

Janvier 1993 : Il existe une cinquantaine de serveurs http dans le monde. Le CERN lance la version alpha du premier browser graphique pour XWindows et Macintosh.

Février 1993 : Marc Andreesen édite la première version du browser Mosaic par le NCSA. Elle fonctionne sous XWindows UNIX.

Octobre 1993 : NCSA lance la première version des browsers Mosaic sous Macintosh et PC Windows.

Mars 1993 : Andreessen et Clark (le fondateur de Silicon Graphics) s'unissent pour développer Netscape.

Juillet 1993 : Le Cern et le MIT puis l'INRIA créent le WWW Consortium pour guider à la normalisation du Web.

Octobre 1994 : Netscape est lancé en beta test sur PC, Macintosh et XWindows.

Février 1995 : 4 millions d'utilisateurs de Netscape : 75% des browsers sont des Netscape.

Mai 1995 : Microsoft annonce la distribution de Spry, un browser sur les versions de Windows 95.

Novembre 1995 : Netscape sort la version 2.03b de son logiciel, qui devient opérationnelle, supporte les News, le courrier (envoi et lecture) et supporte le langage JAVA.

Décembre 1995 : Microsoft lance sa version Internet Explorer 2.0.

Mars 1996 : Microsoft annonce que la version d'Internet Explorer 3.0 supportera Java, JavaScript, les liens OLE2, les Frames. La guerre Microsoft-Netscape est déclarée.

Octobre 1996 : La guerre Internet Explorer 3.0 vs Netscape 3.0 fait rage. Microsoft, à grand renfort d'annonces sur sa technologie Active X, se rapproche à grands pas de la technologie Netcape One.

Décembre 1996 : Tout le monde ne parle plus que des versions Netscape 4 et de Office 97, qui transforment votre PC en un navigateur. On pense que les systèmes d'exploitation de demain seront à base de navigateur. Les machines Java arrivent.

On oublie juste que les utilisateurs ne suivent plus, ne chargent plus les dernières versions qui font plusieurs dizaines de méga octets et que les sociétés en sont encore à Windows 3.11.

Septembre 1997 : Le Web est stabilisé dans sa technologie, du moins provisoirement. La loi antitrust américaine demande à Microsoft de retirer son navigateur des versions Windows 98.

Octobre 1997 : Suite à l'arrêt des Chroniques de Cyberie qui était un des Web de contenu les plus anciens dans le monde francophone, plusieurs centaines de sites ferment leur porte pendant une semaine. 2500 signatures marquent la première grève du Web, elle est partie de France.
Janvier 1998 : Netscape annonce que les versions de son Navigateur sont toutes libres d'utilisation. Netscape compte sur la communauté des développeurs en fournissant les sources de son navigateur. Netscape licencie en même temps 400 personnes. Personne ne souligne encore la nouvelle donne d'Internet dans l'économie de la fabrication des logiciels. La grande déferlante des logiciels contributifs changera-t-elle les donnes de l'économie informatique de demain?
Janvier 1999 : Pendant que se déroule le procès antitrust de Microsoft, la Société Netscape est vendue à Sun et AOL.

2000 et après...
L'usage du Web s'est généralisé tant en ce qui concerne la recherche que le transfert d'information.

6.- LE COURRIER ÉLECTRONIQUE

6.1.- Définition
Le courrier électronique est l'outil le plus répandu, d'abord dans l'Internet des entreprises, puis pour le particulier. Il permet d'acheminer des notes courrier entre personnes éloignées. Le nom anglais e-mail est resté dans le langage, et les utilisateurs parlent de leur adresse e-mail. Nous essaierons d'employer le mot courrier électronique tout au long de ce guide.
L'avantage du courrier électronique sur le téléphone ou sur le fax est considérable. En effet, il permet de joindre un correspondant avec des informations écrites, tout comme le fax, mais qui peuvent être recopiées dans un document en mode texte. Par rapport au téléphone, les courriers électroniques permettent d'aller droit à l'essentiel, évitent les aléas des répondeurs et permettent de laisser des traces écrites. En outre, vous pouvez lire votre courrier de n'importe où dans le monde lors de vos déplacements, ce qui en fait un des meilleurs moyens de joindre un correspondant.
Les logiciels de courrier électronique permettent d'envoyer des documents attachés à la note principale. Ainsi, par le courrier, les utilisateurs d'Internet peuvent échanger des fichiers non ASCII (documents Word, photos, logiciels etc.).

6.2.- Principes
Chaque connecté à Internet possède une ou plusieurs adresses de courrier Internet. On les appelle adresse e-mail ou adresse électronique et le premier réflexe à avoir lorsque vous discutez avec un connecté est de lui donner votre adresse e-mail ou de lui demander la sienne.
Ces adresses sont de la forme : nom@organisation.domaine.

6.3.- Envoyer et recevoir des courriers
Un courrier électronique contient toujours :

• l'adresse du destinataire,
• le sujet du mail parfois appelé aussi objet du courrier,
et de façon optionnelle :
• les lignes correspondant au contenu du courrier électronique. Ce contenu est en ASCII, mais il peut avoir un attachement (c'est-à-dire un fichier qui peut être de n'importe quelle forme : ASCII, Word, son, etc). Le dernier point n'est pas supporté par tous les logiciels de courrier électronique.
Lorsque vous interrogez votre boîte aux lettres électronique, vous rapatriez tous les courriers qui se trouvent sur votre serveur de courrier. Lorsque vous expédiez un courrier à quelqu'un, ce courrier sera envoyé dans la boîte aux lettres de votre destinataire, jusqu'à ce que celui-ci lise son courrier. Il faut remarquer que la boîte aux lettres de votre correspondant peut être située sur son ordinateur, s'il possède un ordinateur connecté au sein d'une entreprise et qu'il dispose d'un logiciel serveur de courrier, sur un serveur de courrier, dans le cas contraire. Ce deuxième cas est le plus fréquent et il est donc important de vous souvenir que le courrier électronique sera stocké sur un serveur tant que vous ne lisez pas votre courrier.

L'expéditeur trouve dans son logiciel de courrier un champ cc: (Carbon Copy). Ce champ est réservé à une liste d'utilisateurs qui recevront le courrier en copie. Dans cette liste de destinataires, chaque adresse sera séparée par une virgule. Dans le même ordre d'idées, un champ bcc: (Blind Carbon Copy) permet de donner une liste de destinataires, mais à l'inverse du champ cc:, chacun des récipiendaires n'aura pas connaissance de la liste des autres lecteurs de ce même courrier. Si vous utilisez une liste de distribution, c'est-à-dire une liste de plusieurs adresses, c'est dans ce champ qu'il faut mettre la liste, évitant à l'ensemble des récipiendaires de trouver la liste de tous les destinataires. On trouve également un champ attachment dans lequel l'expéditeur peut donner un nom de fichier qui sera expédié en même temps que le courrier. Ceci vous permet d'envoyer à votre correspondant un programme, un message sonore, une séquence vidéo, des images.
Il existe un champ reply-to qui permet de donner l'adresse de la personne à qui le courrier sera renvoyé après utilisation de la commande reply. Si ce champ est vide, c'est l'expéditeur du message qui verra la réponse adressée.
Le dernier champ qui s'étale sur plusieurs lignes est en fait le corps du message.
Enfin il est bon de savoir que certains de ces champs sont remplis par le système automatiquement, d'autres doivent l'être par vous. Les champs From, Sender, Message ID, Date sont remplis automatiquement. Il est obligatoire que ce champ soit en ASCII, et il est recommandé qu'il ne comprenne pas de caractères accentués.
Si vous désirez utiliser des caractères accentués vous devez :
- soit configurer ou vérifier que votre logiciel supporte le type d'encodage 8 bits MIME,
- soit remplacer les caractères accentués par leur lettre suivie d'une apostrophe comme ceci : é s'écrit e'.

Il arrive encore que les serveurs de courrier limitent la taille maximum des envois ou qu'ils décomposent les gros messages en plusieurs parties. Ainsi, lorsque vous recevez un courrier multiparties avec un fichier éclaté : doc1/3, doc2/3, doc3/3, il vous faudra recomposer le fichier éclaté en un fichier unique.

Les courriers en erreur

Quand vous avez rempli une adresse d'un destinataire qui se trouve être erronée, le serveur de courrier vous renvoie le courrier avec la raison du refus. En général, vous trouverez dans la zone sujet et dans le corps du courrier la raison du refus, que le domaine ne soit pas valable ou que la personne soit inconnue dans le domaine par exemple. Heureusement le serveur vous renvoie le message que vous aviez expédié, vous permettant ainsi de le réexpédier sans avoir à le reformuler.

6.4.- Listes de distribution

Les personnes qui partagent un centre d'intérêt peuvent se rassembler pour en discuter. Il existe plusieurs méthodes (IRC, les News, les Listes de distribution, les Listservs, les pages Web). Les Listes de distribution contiennent un certain nombre d'adresses et, lorsque vous écrivez à cette liste, tous les destinataires recevront votre courrier.

Vous pouvez, sans le savoir, être inscrit dans certaines listes de distribution, soit par votre fournisseur Internet, soit par l'administrateur réseau de votre entreprise qui les utilise pour vous prévenir des éventuelles informations qui peuvent vous être utiles. Si vous voulez répondre à un tel courrier, faites attention à bien répondre à l'auteur et non pas à l'ensemble des récipiendaires.

6.5.- Echange de fichiers entre utilisateurs

Ce chapitre explicite les différents points à connaître pour envoyer des fichiers par le courrier électronique. Ces différents modes d'emploi s'appliquent aux courriers directs entre deux utilisateur ou aux courriers transitant par un serveur (de News par exemple).

6.5.1.- Compression des données

Comme dans tous les cas de transfert de données, si vous voulez expédier un fichier volumineux, vous devez le compresser pour que celui-ci prenne le moins de place possible. La notion de volumineux est relative puisque, pour une ligne à 2400 bps, un fichier de 100 Ko est gros, alors que pour une ligne à 56 Kbps, c'est un fichier de 1 Mo qui est volumineux. Donc, sauf dans le cas de fichiers de moins de 50 Ko, la compression des données s'impose.

Lorsque vous communiquez directement avec un utilisateur, vérifiez bien quel est son système d'exploitation afin de vous assurer qu'il sera à même de décompresser les données que vous lui envoyez. Référez-vous au chapitre sur la compression de données du module ftp, pour de plus amples informations.

Il n'est pas évident que votre correspondant dispose d'un logiciel de décompression de données, donc, avant de faire des expéditions par le biais du courrier, vérifiez également ce point.

uuencode/uudecode

Au début de sa conception, le courrier électronique ne permettait d'envoyer que des fichiers ASCII avec un codage à 7 bits. Les 7 premiers bits servent à coder les principaux caractères mais pas les caractères accentués, qui peuplent nos textes ASCII francais. Ces caractères accentués sont en effet codés sur le 8ème bit qui n'est pas compris par le courrier électronique.

La deuxième fonction de uudecode/uuencode est de tronçonner les fichiers de gros volumes vers une succession de fichiers plus petits. Ainsi le fichier manuel.htm sera décomposé en un fichier manuel1.uue manuel2.uue manuel3.uue. De façon générale, vous utiliserez un programme uudecode pour passer de manuel1.uue manuel2.uue vers manuel.htm.

Document attaché (MIME)

Cette possibilité a été développée sous l'appellation MIME (Multi-purpose Internet Mail Extensions). Les logiciels capables de recevoir ou d'expédier des courriers avec des documents attachés utilisent des codes MIME pour signaler la façon dont ces documents sont codés.

Pour envoyer un document avec attachement, il suffit généralement de faire glisser l'icone du document depuis le gestionnaire de fichier vers votre logiciel de courrier ou de trouver le menu attachment dans le menu message attach document dans le cas d'Eudora.

6.5.2.- Protocoles courrier

Vous verrez souvent les noms SMTP, POP3 apparaître dans vos logiciels de courrier. Sans vous expliciter le détail de chacun de ces protocoles, nous allons en donner les grandes lignes.

SMTP

SMTP (Simple Mail Transport Protocol) regroupe tous les protocoles concernant le courrier électronique sur Internet. Le protocole SMTP est un protocole point à point, c'est-à-dire qu'il met en communication deux serveurs : celui de la personne qui envoie un courrier et celui de la personne qui le reçoit. Ces serveurs sont des machines qui sont chargées de la gestion de votre courrier et de celui de vos collègues. Ainsi si votre nom est dupont@imaginet.fr, vous serez géré par un serveur de courrier qui aura un nom (sans doute mail.imaginet.fr). Ce serveur de courrier aura la charge d'acheminer votre courrier vers le serveur de votre destinataire. Il se peut que la liaison point à point ne concerne qu'un serveur, quand une personne vous envoyant un courrier est gérée par le même serveur que le vôtre. Le protocole SMTP spécifie le format des adresses des utilisateurs, les champs de vos courriers (from: to: etc.), les possibilités d'envois groupés, la gestion des heures.

Si vous êtes connecté chez vous de façon intermittente, votre serveur utilisera SMTP pour recevoir votre courrier et vous utiliserez POP3 pour lire les courriers qui vous attendent sur le serveur. Pour expédier votre courrier, selon la version de votre logiciel, vous utiliserez SMTP directement ou une procédure extension de POP3 pour demander à votre serveur d'envoyer votre courrier.

POP3

Le protocole POP3 a été conçu pour vous permettre de récupérer votre courrier sur une machine distante quand vous n'êtes pas connecté en permanence à Internet. Voilà pourquoi dans votre logiciel de courrier vous devrez donner l'adresse de votre serveur POP. Le protocole POP gère l'envoi de messages identifiés par une clé et un argument, ainsi que la réception de messages d'erreur (ERR) ou d'acquittement (OK).

Le protocole POP gère les messages suivants :

LIST donne le nombre de courriers présents sur le serveur avec leur numéro,
RETR numéro récupère le courrier numéro en attente sur votre serveur,
DELE numéro détruit le courrier numéro,
NOOP vérifie la connexion,
LAST récupère le dernier message arrivé sur le serveur,
QUIT quitte la session et en autorise une autre.

D'autres messages comme TOP (pour voir les titres des messages) ou APOP (qui permet de ne pas envoyer de multiples mots de passe dans le cas des connexions périodiques) sont en marge du protocole.

L'envoi de messages (par les extensions XTND XMIT) n'est pas non plus supporté par le protocole POP3 de base. Le protocole POP gère l'authentification, c'est-à-dire la vérification de vos nom et mot de passe. Il bloque également votre boîte aux lettres pendant que vous y accédez, ne permettant pas à une autre connexion d'accéder en même temps à votre courrier.

Notons que le protocole POP3 n'est pas sécurisé et que, en vous connectant en telnet sur le port 110, vous pouvez voir les messages de réception en clair.

Toutes ces informations permettent de comprendre certaines options de vos logiciels, mais elles ne sont pas implémentées sur tous. Au niveau d'une utilisation normale, vous n'aurez en principe qu'à connaître l'adresse de votre serveur POP pour renseigner votre logiciel.

IMAP

IMAP (Interactive Mail Access Protocol) est un autre protocole moins utilisé que POP, parce que moins répandu, mais qui offre plus de possibilités.

Il gère plusieurs accès simultanés, ainsi que plusieurs boîtes aux lettres sur le serveur et il permet les recherches de courrier selon critères. Il est donc plus riche, mais étant plus complexe, il est moins utilisé.

6.5.3.- Sécurité

Le courrier électronique n'est pas sécurisé au niveau de la confidentialité car les messages sont stockés en clair sur votre serveur de courrier. De plus, vous n'êtes jamais sûr à cent pour cent que votre courrier va atteindre son destinataire. Les solutions pour y remédier consistent en deux points :

- crypter vos messages,
- demander un acquittement.

Cryptage

Il existe plusieurs façons de crypter les messages mais, quelle que soit la méthode retenue, le principe est le même. Il consiste, en effet, à coder le corps du courrier avec une ou plusieurs clés de protection. Le cryptage par un seul mot de passe suppose que vous donniez le mot de passe à vos destinataires par un canal de communication sûr et qu'eux-mêmes ne laissent pas traîner sous les regards indiscrets ce mot de passe. Si on suppose que vos destinataires auront plusieurs correspondants, ils devront connaître un certain nombre de mots de passe.

Un moyen plus sûr est de disposer de deux clés, l'une dite publique et l'autre dite privée. La clé publique sera utilisée pour coder le message et la clé privée sera utilisée pour le décoder. Si l'expéditeur utilise sa clé privée pour coder le message, le destinataire devra entrer la clé publique pour le décoder, on parlera dans ce cas de signature électronique ou digitale. Tout ceci permet de s'assurer que les courriers électroniques n'ont pas été modifiés ou lus pendant leur transport.

Les logiciels de cryptage offrent des annuaires de clés de vos correspondants, accessibles par un mot de passe et une option offrant une visualisation uniquement des premiers caractères de ces clés, ceci afin d'abriter les clés des regards indiscrets.

PGP (Pretty Good Privacy) est une solution de cryptage mise au point par Philip Zimmermann. PGP est devenu très vite le standard de cryptage des messages, parce qu'il est gratuit, d'une part et parce que le MIT soutient le protocole. Standard ne veut cependant pas dire unique, puisqu'il existe d'autres systèmes. Comme d'autres logiciels de cryptage, PGP utilise deux clés, l'une publique et l'autre privée. La clé publique permet un premier niveau de sécurité dans la mesure où seules les personnes qui connaissent cette clé peuvent vous écrire ou vous lire. PGP utilise un code de validation en fin de codage qui permet de valider la cohérence du message crypté. Mais c'est la clé privée que votre correspondant devra posséder qui permettra la parfaite sécurité du protocole. S/MIME définit un protocole de messagerie sécurisée qui est de plus en plus utilisé par les programmes récents d'échange de courrier.

6.5.4.- Netiquette

- Ne pas encombrer votre boîte aux lettres.

- Vérifier vos courriers régulièrement si vous en recevez beaucoup. Si vos courriers résident sur un autre disque dur que le vôtre, une fois qu'ils sont lus, ne les conservez pas indéfiniment mais détruisez-les ou rapatriez-les.
- Sachez que les courriers électroniques ne sont pas confidentiels. Ils vont transiter sur le réseau et être stockés sur des disques durs jusqu'à ce que vous les lisiez donc il est en principe facile à n'importe quel administrateur de lire un de vos courriers électroniques. Evitez les informations ultra-confidentielles.
- Ne vous abonnez pas à trop de listes de distribution. Sachez que si, par une soirée d'hiver, vous vous abonnez à une dizaine de listes de distribution, il peut bien vous arriver 100, 200 ou 1000 courriers sous vingt-quatre heures. Abonnez-vous progressivement à vos listes de distribution préférées.
- Soyez sûr de vos adresses. La plupart du temps, les envois erronés sont sans incidence mais si vous envoyez un très gros courrier (notamment avec des attachements), une erreur peut vous coûter de longs temps de connexion.
- N'oubliez pas le champ sujet. Le champ sujet est celui qui est lu en premier lieu par vos correspondants pour identifier votre message. Ne l'oubliez pas et, de plus, mettez un texte concis et le plus explicite possible. En outre évitez l'emploi de caractères dollars car ces caractères sont parfois éliminés par les filtres anti-spam.

7.- LES NEWS - USENET

7.1.- Présentation des News

Les News sont des forums fédérés par thème, où, pendant une durée de temps donnée, tous les courriers envoyés sont conservés. Ainsi sur un forum réservé aux cerfs-volants, les questions des uns sont envoyées par mail et, quelques heures plus tard, d'autres peuvent y répondre. Les News sont de formidables réservoirs d'informations vivantes sur un sujet. Alors que les courriers électroniques entre individus ou au travers de groupes de diffusion sont stockés dans les boîtes aux lettres de chacun des correspondants, les News ne sont pas envoyées à tous les utilisateurs. Elles sont consultées par ceux qui sont intéressés par leur sujet.

Une question ou un article comprend un titre (title) et un corps (body). Les réponses reprennent le titre précédé des trois caractères Re: (pour réponse).

Un ensemble Article + Réponse est appelé un Thread et est regroupé pour faciliter la lecture suivie d'une discussion. En premier niveau, votre logiciel de lecture de News ne vous présentera peut-être que les titres de Thread sans visualiser la liste des réponses.

Les grandes questions souvent répétées sur un sujet donné sont concentrées dans des fichiers FAQ (Frequent Asked Questions) qu'on retrouve une fois par mois sur le forum concerné. La traduction française des FAQ est Foire aux Questions. Souvent, on retrouve une page Web associée à un forum, qui présente en format HTML le fichier FAQ.

Chaque forum est appelé en anglais newsgroup, chaque article d'un newsgroup est appelé une News. Les termes français les plus proches sont forum et article mais il est probable que si vous employez ces termes, peu de gens, sur Internet, vous comprennent. La tendance actuelle est d'utiliser le mot nouvelle mais le terme n'est pas encore entré dans les moeurs et tout le monde parle de News.
Certains groupes sont dits modérés lorsque les articles qui sont envoyés sont contrôlés par un ou plusieurs responsables (appelés modérateurs) qui pourront accepter ou refuser la publication de l'article.
Les différents newsgroups doivent être au nombre de 17 000 par le monde. Mais il se peut que le serveur de News qui vous alimente ne possède qu'une partie de ces News. On peut accéder aux News par un logiciel de lecture de News mais également par un lecteur de Web. Les adresses URL de News par logiciel Web sont de la forme : news:adressenews. Ainsi l'adresse URL news:misc.test correspond au groupe misc.test.

7.2.- Fonctionnement

Des messages sur l'Usenet sont acheminés de serveur de News en serveur de News en utilisant des protocoles spécifiques aux News. Un serveur de News garde tous ses messages sur un disque dur, que chaque connecté peut aller consulter. Chaque serveur de News compare avec un autre serveur de News la liste de ses articles dans chacun des groupes et les serveurs s'échangent les nouveaux articles. Ces comparaisons ont lieu chaque jour entre les serveurs et cela provoque des millions d'échanges sur l'Internet.
Comme le nombre de groupes est important, les utilisateurs ne retiennent que les groupes qui les intéressent. Ainsi, chaque connecté à la News conserve-t-il la liste des groupes auxquels il est "abonné". Cette notion d'abonnement est bien sûr indépendante de toute notion d'abonnement payant. De plus, comme le chargement du contenu de tous les articles prendrait du temps, les logiciels de consultation de News ne chargent que les titres des News. C'est au connecté de charger les corps des messages qui l'intéressent.
Le protocole de gestion des News est NNTP (News Network Transfer Protocol), il gère aujourd'hui des connexions permanentes avec ses serveurs voisins, mettant à jour instantanément chaque nouveauté. Notons que le protocole NNTP connaît comme unité de traitement l'en-tête du courrier composant la News.

7.3.- Une histoire de USENET

Au début d'ARPANET (l'ancêtre d'Internet), les premiers échanges de courriers électroniques laissaient présager de l'intérêt que pouvaient avoir certains courriers pour d'autres lecteurs que les correspondants originels. De plus, les discussions sur des thèmes se répétaient entre plusieurs correspondants sans que ceux-ci soient informés des travaux de leurs confrères. L'USENET fut créé en 1979, après la version UNIX qui supportait UUCP (UNIX V7).

Tom Truscott et Jim Ellis, deux étudiants de l'université de Duke, ont jeté les bases de l'échange des données sur l'USENET. Steve Bellovin, de l'Université de Caroline du Nord, mit au point la première version des News en Shell script et l'installa sur les sites de UNC et de Duke. Dans les années 1980, le shell script fut ré-écrit en langage C. Et c'est Steve Daniel qui rendit publique la première version C. Tom Truscott fit les premières modifications pour donner la version 1 des News.
La version B naquit en 1981 à l'Université de Berkeley, Mark Horton et Matt Glickman réécrirent le logiciel en implémentant de nouvelles fonctionnalités. 1982 vit la version 2.1, première release non Beta Test, puisque les précédentes furent considérées jusque là comme des beta releases. Rick Adams, du Centre d'Etudes Sismiques, fut l'auteur de la version 2.10.2 en 1984. Cette version inclut le concept de groupes modérés. La version 2.11 de 1986 supporta les compressions de données dans les News. La version actuelle est la version 2.11 avec le patch 19.
Une version Beta nommée C fut ensuite développée à l'Université de Toronto, dans le but de ré-écrire les couches basses du protocole afin d'améliorer la vitesse de lecture des News, la mise à jour des nouveaux articles et la gestion de l'expiration de News.

7.4.- Netiquette des News

Toutes les règles de savoir-vivre de Usenet (c'est-à-dire des News) sont appelés Usenet Netiquette. La Netiquette, rappelons-le, prend en compte les aspects de courtoisie sur Usenet mais traite également de l'optimisation de l'utilisation des News, afin de ne pas surcharger les différents serveurs. Suivre la Netiquette permet d'assurer la survie d'Internet, pensez-y!

Signatures

A la fin de chaque article figure une signature. Cette signature doit comprendre votre adresse électronique, votre numéro de téléphone et votre adresse postale. Elle permet à une personne lisant les News de vous contacter si votre article l'intéresse.
Notons que, dans la signature, certains mettent des graffiti, des notes philosophiques (comme par exemple : un mathématicien est une personne qui transforme du café en théorème). En général, cette signature est paramétrable dans votre logiciel et automatiquement insérée à la fin de votre article. Quatre lignes doivent suffire à votre signature et aux graffiti que vous mettez en sus. En effet, plus génère du trafic réseau. Imaginez un utilisateur chargeant sur une ligne modem à 2400 bits par seconde, un groupe de 300 articles comprenant chacun 100 caractères de trop et vous lui faites perdre 2 minutes. Et pensez qu'il existe en moyenne 1000 connectés par serveur et 10 000 serveurs de News dans le monde.
L'oubli d'une signature est moins grave que de renvoyer une deuxième fois l'article sur la News.

Poster des messages personnels
N'utilisez pas les News pour poster des messages personnels, à une autre personne qui lira votre prose au travers des News. Si votre courrier électronique pose un problème, résolvez-le plutôt que de passer par cette combine.

Les courriers
De même, évitez de mettre dans les News une réponse à un courrier qui ne figurait pas dans les News, c'est une règle de politesse élémentaire à respecter envers votre correspondant. Sans parler de votre réaction si vous découvrez dans un groupe sans que vous en soyez informé, votre prose commentée, critiquée ou raillée.

Messages de test
Beaucoup de nouveaux utilisateurs veulent essayer de poster un message pour voir comment les News fonctionnent. Cette volonté de tester est légitime mais ne doit pas être faite vers des groupes normaux. Il existe un certain nombre de groupes pour cela, appelés des groupes de test, qui permettent de tester un envoi, de tester sa dernière signature ou tester un nouveau logiciel de lecture de News.
Citons les groupes :

alt.test [news:alt.test]
gnu.gnusenet.test [news:gnu.gnusenet.test]
misc.test [news:misc.test]

Certains de ces groupes vous renverront même un courrier en retour pour vous dire que le vôtre a été bien reçu.

Sommaires
Vous verrez parfois des auteurs d'articles promettre qu'ils feront un sommaire des réponses à la question qu'ils posent. Les lecteurs de News doivent donc savoir que leurs réponses sont susceptibles de faire partie d'un compte-rendu ou d'un sommaire. Quand vous créez un sommaire des réponses, essayez de faire un document le plus homogène possible. Evitez de mettre juste l'ensemble des réponses pêle-mêle dans un gros fichier. Prenez le temps d'éditer les messages dans une forme qui contienne l'ensemble des informations intéressantes.
Parfois aussi, des lecteurs vont répondre de manière anonyme, soit parce qu'ils répondent de leur travail, soit parce que leur réponse comprend des opinions qu'ils ne souhaitent pas exprimer publiquement. Respectez leur signature en ne reproduisant pas la copie de leur adresse électronique.

Citation
Dans un courrier en réponse, beaucoup de logiciels lecteurs de News permettent de citer l'article original par un symbole préfixant chaque ligne du message original. On trouve souvent le caractère > par exemple. Ceci est bien commode pour répondre point par point à un article paru dans une News, mais dans le cas où cet article est long et où vous n'avez que quelques commentaires à faire, ne citez pas l'ensemble

de la News. Préférez plutôt prendre une partie de l'article que vous recopiez dans votre réponse.

Crossposting

Le nom du newsgroup (champ Newsgroups:) n'est pas limité à un seul groupe. Un article peut être posté dans plusieurs groupes simultanément. Les différents noms de groupes de News sont séparés par des virgules et aucune limite théorique en nombre de groupes n'est fixée. Cependant, 3 ou 4 doivent être considérés comme une limite maximale à ne pas franchir.

Evénements récents

Poster des réactions à des événements récents (accident d'avion, décès etc..) n'est pas une bonne chose, en effet le temps que la News se propage, tout le monde est informé par les médias traditionnels et votre information risque bien d'être obsolète ou inutile à sa lecture. Il faut préférer les moyens temps réel pour ce genre d'information (IRC par exemple).

La qualité de vos questions

La façon dont vous écrivez et dont vous vous présentez dans la News est une chose importante. Faire vingt fautes par ligne ne vous positionne pas comme un leader de pensée et même si vous écrivez dans une langue qui n'est pas votre langue maternelle, relisez et contrôlez vos écrits avant d'appuyer sur le bouton envoi (ou send) de votre logiciel de News. Essayez d'être clair et concis dans la formulation de vos questions, sinon vous n'obtiendrez pas de réponse. En effet, les questions nécessitant une réponse longue de plusieurs volumes décourageront les personnes qui pourraient vous répondre. Par exemple, les questions " Comment me servir de mon PC? " ou " Est-ce que Dieu existe? " sont des questions un peu trop vastes pour espérer une réponse. Dans les exemples précédents, préférez les questions " Où pourrais-je trouver une documentation d'utilisation de PC? " ou " Quels sont les Web spécialisés dans les questions théologiques ? ".

Mettre un sujet utile

Le sujet est la ligne qui apparaîtra pour informer les lecteurs du contenu de l'article. C'est à partir de ce sujet que les lecteurs trouveront un intérêt à lire ou à ignorer votre prose. Il y a de bonnes formulations et de mauvaises, par exemple :

oui

    Subject: Où trouver une carte de Paris ?

oui

    Subject: Que penser de la peine de mort ?

non

    Subject: J'arrive pas à trouver une carte de Paris.

non

    Subject: Je suis pour la peine de mort.

En effet, il est plus intéressant de poser une question générique que de parler de vos problèmes.

Ton du message
Les News ne transportent pas les intonations de la voix, les inflexions ou le ton des phrases. Personne ne sait que vous maniez un humour noir ou que votre mail comprend des contrepèteries. Donc préférez un ton sobre et passe-partout, évitez les lettres majuscules et les dix points d'exclamation en fin de chaque phrase.

7.5.- Les FAQ (Frequent Asked Question - Foire aux Questions)
Les groupes incluent généralement une FAQ qui donne les réponses aux questions les plus fréquentes. Ces FAQ sont d'abord utiles pour un nouveau lecteur qui veut connaître les grandes lignes d'un sujet. Il lui suffit de lire les réponses aux questions que le plus grand nombre de lecteurs se sont posées.
Normalement, les FAQ sont remises à jour tous les mois. Il faut également savoir que le groupe news.answers [news:news.answers] contient toutes les FAQ d'Internet.
Il est préférable, avant de poser une question sur un sujet, de vérifier si la réponse ne figure pas dans les FAQ.

8.- FTP (File Transfer Protocole)

8.1.- Utilité
FTP (File Transfer Protocol) est le premier outil qui a été mis à la disposition des utilisateurs pour échanger des fichiers sur Internet ou TCP/IP. En utilisant FTP, vous serez client d'un modèle client/serveur et vous vous adresserez à un serveur de fichier par ftp.
Des milliers de serveurs sont connectés sur l'Internet et proposent des trésors de logiciels shareware ou freeware, accessibles au public. Vous trouverez sur un serveur ftp des logiciels d'arbres généalogiques, des logiciels d'échecs, des logiciels de comptabilité, de traitement de textes. Vous trouverez des poésies ou des romans noirs pour vos nuits blanches. Vous trouverez toutes les réponses aux questions que vous vous posez dans des fichiers FAQ ou les explications sur le protocole TCP/IP dans les RFC.
Il est important de connaître une convention d'utilisation importante au sujet de ftp : en théorie, on ne peut se connecter sur un site par ftp que si on possède un compte et un mot de passe sur ce site ; en pratique, l'usage veut que tous les serveurs présents sur l'Internet aient un compte anonymous. Le mot de passe de ce compte anonymous n'est pas mis en place mais il est demandé de mettre, dans le champ mot de passe, son adresse électronique.
On trouve plusieurs implémentations de logiciel ftp, certaines rudimentaires avec des commandes manuelles, d'autres avec des interfaces graphiques. Les logiciels ftp

sont remplacés aujourd'hui par les lecteurs de Web, du moins pour la lecture de fichiers distants. Pour l'écriture de fichiers distants, les lecteurs de Web permettent aujourd'hui de remplacer ftp si les serveurs http sont pourvus d'une interface de chargement adéquate.

8.3.- Compression et format des données

8.3.1.- Généralités

Lorsque vous vous connectez sur un serveur, les données stockées sur ce serveur sont très souvent compressées, la compression des données servant à réduire leur espace de stockage. Il existe de nombreuses techniques de compression des données. Pour être très simpliste, disons que, pour compresser un fichier, les logiciels peuvent, par exemple, remplacer les répétitions de caractères par le caractère suivi par le nombre de fois qu'il est répété. On comprend que chaque séquence 0000000000000, si elle est remplacée par 0 13 (pour indiquer 0 treize fois), peut diviser la taille du fichier qui la contient de façon notable. Ceci est d'autant plus vrai pour les fichiers images, où l'on retrouve dans un fond d'écran, par exemple, une suite de points noirs. Les fichiers compressés, pour le plus grand bien des serveurs ftp et des lignes réseaux, devront être décompressés sur votre ordinateur.
La problématique qui se posera à vous est la suivante : chaque ordinateur a son propre format de compression et il n'est pas évident de pouvoir tout décoder sur une plate-forme donnée. Vous devrez vous procurer les logiciels de décompression vous permettant de lire le plus grand nombre de fichiers sur votre ordinateur. Le suffixe du nom du fichier renseigne souvent sur son mode de compression. Regardons ces différents modes de compression :

txt l TXT :	fichiers textes (ASCII) [non compressés]
ps l PS :	fichiers PostScript destinés à être imprimés
doc l DOC :	fichiers Microsoft Word [non compressés]
Z :	fichiers compressés à la mode UNIX par compress : utilisez uncompress
z :	fichiers compressés à la mode Unix par une commande pack : utilisez unpack
ZIP :	fichiers compressés à la mode PC
gz :	fichiers compressés par le compresseur GNU gzip : utilisez gunzip
Hqx l hqx :	programmes compressés sur Macintosh : utilisez StuffIt
shar :	compressés par shar sous UNIX : utilisez unshar
tar :	fichiers assemblés sur UNIX par tar : utiliser untar
Sit l Sit :	format Macintosh : utilisez StuffIt
ARC :	compressés sous DOS par ARC
EXE :	peut-être un programme compressé qui s'auto-décompresse en le lançant Attention cela peut aussi contenir un virus!

Les fichiers les plus courants sont les fichiers ZIP sur PC, HQX sur Mac et gz sur Unix. On arrive à trouver des logiciels permettant de décompresser un fichier n'ayant pas été compressé sur le même type de matériel.

8.3.2.- Les logiciels de décompression

Plusieurs programmes (souvent gratuits) prennent en charge la compression/décrompression des fichiers.

Windows : Le logiciel de référence est winzip http://www.winzip.com/Page

Macintosh, linux et Windows : Utiliser par ex. Stuffit http://www.stuffit.com/.

8.4.- Virus

Les virus sont des morceaux de code qui ont pour objet de provoquer des anomalies de fonctionnement graves ou amusantes sur votre ordinateur. Pour mériter leur nom de virus, ils se propagent d'ordinateurs en ordinateurs, infectant de proche en proche les différents matériels rencontrés. Les personnes qui en écrivent sont en général de jeunes informaticiens qui, après quelques années d'études ou de pratique de l'informatique, pensent qu'ils ont découvert le monde et que le monde leur appartient.

Il est très facile d'écrire un virus qui endommagera un disque dur ou un réseau. C'est une chose que peut faire n'importe qui, après un mois de formation. Il est difficile de comprendre pourquoi ceux qui ont recours à de telles méthodes tirent une telle fierté de leur acte terroriste.

Il faut être conscient que vous prenez un risque quand vous exécutez un logiciel chargé et ce au moment où vous l'exécuterez.

Enfin, vous voyez que les programmes d'extension .exe peuvent s'auto-décompacter. Une âme malveillante pourrait aussi mettre un programme de formatage de disque dur à la place (en pensant qu'il est très fort d'avoir eu une si brillante idée).

Donc sauvegardez vos données si vous vous connectez sur Internet...

Pour ma part, le seul virus dont j'ai été victime me fut transmis par une disquette de démonstration distribuée par un magazine. Je n'en ai jamais vu sur Internet. Donc la prudence est de mise mais pas la psychose.

9.- TELNET

Telnet (TErminaL on NETwork) est le protocole permettant de se connecter sur une machine distante en tant qu'utilisateur. Cela vous donne la possibilité d'accéder depuis chez vous à votre ordinateur professionnel. Cela permet à des professionnels de maintenir un système distant de plusieurs milliers de kilomètres. Telnet suppose que la machine sur laquelle vous vous connectez soit un serveur. La machine depuis laquelle vous vous connectez est un client Telnet. Souvent, c'est par le Web que vous accéderez à un numéro Telnet en cliquant sur un lien (URL) telnet ou en donnant l'adresse URL dans votre lecteur de Web. Pour utiliser telnet vous ferez toujours une connexion sur une machine d'un domaine. Il est important de garder présent à l'esprit que, lorsque vous accéderez à un système par telnet, vous serez généralement connecté en mode UNIX, mais un logiciel simple d'utilisation (genre logi-

ciel Minitel) vous permettra de faire des opérations par sélection de numéros dans un menu. Généralement, le système vous demandera un nom et un mot de passe. Souvent des questions sur le terminal que vous utilisez vous sont posées, en général le mode VT100 est le mode passe-partout.
Ne restez pas connecté sur un serveur Telnet quand vous n'avez plus de requête à formuler ; souvent, en effet, les serveurs telnet ont un nombre d'accès limité, et peut-être que d'autres utilisateurs attendent pour se connecter.

10.- IRC (Internet Relay Chat)

IRC (Internet Relay Chat) est un concept qui date de 1998. Il s'agit d'un protocole qui permet à des utilisateurs de communiquer en direct. IRC permet de discuter à plusieurs dans des forums (canal) ou à deux (en privé). On retrouve autant de canaux que de thèmes, un peu comme dans les News, mais à la différence des News, chacun peut créer un canal qui sera détruit automatiquement dès qu'il sera vide. De façon générale, IRC est utilisé pour discuter de choses sérieuses et de choses moins sérieuses, dans différentes langues. Le pire y côtoie le meilleur puisque chacun peut créer son canal à tout moment.

Pour le fun : les Smileys, quelques exemples

Le principe des Smileys est de représenter la tête d'un bonhomme de façon très schématique avec des signes simples. Pour reconnaître le bonhomme, il faut pencher la tête sur la gauche afin de deviner en :-) les yeux, le nez et la bouche. Voici les Smileys que vous avez à connaître pour comprendre ce que vous racontent vos interlocuteurs :

%-)	je commence à ne plus avoir les yeux en face des trous
'-)	clin d'oeil
:-(	je suis triste
:#)	je suis saoul
:'-(	je pleure
:'-)	je pleure de rire
:(	je suis triste
:)	je suis content
:*	baiser
:*)	baiser sur la joue
:,(	je pleure
:-#I	je porte une moustache
:-{	je porte une grosse moustache
:-(	je suis triste
:-(*)	j'ai mal à la gorge
:-)	je suis content, je trouve ça drôle
:-*	je t'embrasse
:-/	je suis sceptique
:-0	je crie
:-<	je suis très triste
:->	je ris
:-C	je suis mécontent
:-D	je me moque de vous
:-S	je dis n'importe quoi
:-X	très gros baiser
;-)	clin d'oeil

La voix et la vidéo sont maintenant présentes dans les derniers logiciels IRC.

11-. LES ENCYCLOPÉDIES DYNAMIQUES

Le concept d'encyclopédie est maintenant déployé sur le web. Des documents, qui couvrent tous les domaines de la connaissance, sont rassemblés et organisés par des volontaires qui proposent ces pages au public. Cette approche permet d'une part de s'informer et d'autre part de corriger et d'améliorer la qualité des informations car, si lors d'une consultation vous remarquez une erreur ou une omission vous êtes engager à la signaler pour demander une correction. De plus, si vous vous en sentez capable, vous pouvez soumettre des documents sur un sujet qui n'est pas encore traité en vue d'augmenter "dynamiquement" la somme des connaissances reprises dans l'encyclopédie. Cette approche semble promise à un bel avenir et on constate souvent que la qualité des documents proposés est très bonne. L'exemple le plus représentatif de ce nouveau concept est le projet WIKI (cf. http://fr.wikipedia.org). Citons aussi dans le domaine technologique le site "comment ca marche" (cf. http://www.commentcamarche.net/)

C H A P I T R E VIII

Réseau ULg : Aspect pratique

Les machines dont vous disposez sont toutes reliées au réseau de l'Université de Liège. Ce réseau fait partie de l'Internet mondial et nos machines sont donc capables d'accéder directement à la plupart des services disponibles sur ce réseau. Nous allons donner ci-après quelques informations techniques utiles ainsi que certains conseils déontologiques spécifiant la manière correcte d'utiliser ces ressources.

Pour être relié à l'Internet, il suffit de disposer d'un micro-ordinateur, d'un modem ou d'une liaison directe et du **"droit" de connexion et d'un "point d'accès"**. Ce droit dépend de critères déontologiques et commerciaux mal définis et fait actuellement l'objet de discussions délicates au niveau mondial. Le monde économique, qui a perçu l'enjeu commercial de l'Internet, tente de se l'approprier pour le transformer en un service. En Belgique, les universités sont membres de l'organisation BELNET qui assure la gestion du réseau belge destiné aux activités d'éducation et de recherche.

1.- REMARQUES LIMINAIRES

L'accès à l'Internet est un privilège et non un droit. Le fonctionnement général du réseau repose sur un code de bonne conduite librement consenti. IL NE FAUT PAS EN ABUSER sous peine de s'en voir interdire l'accès!
En ce qui concerne le courrier électronique, les messages doivent être rédigés en anglais (sauf si les deux correspondants sont d'accord d'utiliser une autre langue). Leur contenu doit être conforme aux bonnes moeurs et il est interdit d'utiliser le réseau à des fins commerciales ou pour "pirater" des programmes. Sachez aussi que la confidentialité des messages n'est pas assurée.
Lors de transferts de fichiers, il faut limiter le temps de connexion au strict minimum et, si possible, réaliser les transferts de gros fichiers en dehors des heures habituelles de travail. Par exemple, il vaut mieux adresser le réseau américain le matin, puisque, grâce au décalage horaire, les Etats-Unis dorment toujours à ce moment…

2.- NOMENCLATURE DES ADRESSES

Toutes les machines reliées à l'Internet doivent disposer d'une adresse unique. Cette adresse est notée sous la forme de quatre nombres séparés par des points :
Ex : `139.165.120.13`

Cette adresse est obtenue dynamiquement (cf. protocole DHCP) ou attribuée à chaque machine par les responsables du réseau et ne peut alors pas être changée. L'Université de Liège dispose d'un réseau de type "B", ce qui signifie que les deux premiers octets de l'adresse (139.165) ont été attribués par les gestionnaires mondiaux du réseau tandis que les 2 autres sont gérés par les autorités locales.

Pour faciliter l'identification des machines, on associe à chaque adresse un nom mnémonique formé de zones textuelles séparées elles aussi par des points :

Ex : `zero.phe.ulg.ac.be`

Ce nom renseigne la localisation de la machine. Le dernier champ indique le pays (be = Belgique, ch = Suisse ; fr = France, etc…). Notons cependant que les Etats-Unis, attribue à la zone finale un sens différent : edu = université américaine ; com = firme commerciale ; mil = un centre militaire ; gov = un centre du gouvernement). Les autres champs donnent de plus en plus précisément la localisation de la machine :

ac = académique
ulg= Université de Liège
phe=Etudiant en Physique
zero=une des machines.

☞ Pour toutes les machines de l'université, la finale est la même (ulg.ac.be =139.165), le reste dépend du service.

Les adresses dynamiques

Une adresse dynamique peut aussi vous être attribuée pour une période de durée déterminée. Le protocole généralement utilisé est le **DHCP** (Dynamic Host Configuration Protocol). Il s'agit d'un protocole qui permet à un ordinateur qui se connecte pour la première fois à un réseau d'obtenir rapidement et automatiquement (c'est-à-dire sans intervention particulière) les paramètres d'une configuration ad-hoc compatible avec ce réseau. On se voit ainsi attribué une adresses IP dynamique pour un temps donné. Ce fonctionnement est recommandé pour les ordinateurs portables qui peuvent ainsi être connectés sans problème aux différents réseaux (WIFI, hotels, salles de cours, etc...).

Le mode DHCP est cependant déconseillé pour les serveurs, car ils doivent conserver la même adresse IP pour rester accessible.

Plus d'info : http://www.commentcamarche.net/internet/dhcp.php3

3.- LES SERVICES DISPONIBLES SUR LE RÉSEAU

3.1.- L'émulation de terminal

Un terminal est une machine qui est connectée à un ordinateur éloigné par un câble ou par réseau. Il permet de contrôler à distance le fonctionnement de cet ordinateur. Il se compose au minimum d'un clavier et d'un écran. La plupart des ordinateurs actuels peuvent se transformer en terminal grâce à des programmes spécifiques. Différents modes de présentation sont possibles.

• Mode TTY (Télétype)
Il s'agit du mode d'accès le plus élémentaire pour un terminal. Les caractères reçus sont imprimés à l'écran comme s'il s'agissait d'une feuille de papier qui se déroule. Très peu d'options de mise en page sont disponibles et le fonctionnement est séquentiel.

• Mode VT100
C'est le mode d'émulation le plus souvent adopté pour les terminaux non graphiques. Lorsque le terminal fonctionne en mode "plein écran" (c'est-à-dire que la position des lettres est contrôlée par le programme et que l'on peut écrire n'importe où dans un écran qui comprend au minimum 24 lignes de 80 caractères). Le terme VT100 vient du nom d'un des terminaux de la gamme de Digital Equipment qui est devenu un standard de fait.

• Mode IBM 3270
Les terminaux connectés aux anciens ordinateurs IBM fonctionnent selon un mode spécifique dit 3270 (le nom fait référence aux numéros d'identification de ces terminaux dans la gamme IBM). Il s'agit aussi d'un fonctionnement limité au texte dont les instructions de mise en page sont particulières.

• Mode X
Les terminaux X fonctionnent en mode graphique et contiennent habituellement un microprocesseur spécialisé qui gère l'écran. Le mode X permet de disposer d'une interface utilisateur graphique (GUI) sur un terminal distinct de la machine principale. Le protocole de liaison est très complexe et repose sur la notion de messages concis qui sont interprétés par le microprocesseur du terminal pour activer des processus graphiques compliqués comme, par exemple, dessiner une fenêtre ou tracer des formes géométriques. Cette approche réduit le trafic sur les lignes de transmission tout en répartissant la charge de la gestion d'écran entre le processeur principal et celui du terminal. Un terminal X contient souvent plusieurs mégaoctets de programmes et de définitions graphiques (jeux de caractères, etc…).

• Mode “Bureau à Distance “
Le Client “Connexion Bureau à Distance” (remote desktop client) vous permet de vous connecter depuis votre ordinateur à un autre ordinateur équipé d’un serveur “Bureau à distance” (disponible sur toutes les versions récentes de Windows par ex.). On peut alors utiliser les programmes et les fichiers qui se trouvent sur la machine hôte en observant sur sa machine une réplique quasi parfaite de l’écran du serveur, Il est aussi possible d’échanger des informations entre les machines (couper/coller etc...) et d’activer la plupart des périphériques de l’hôte.

3.2.- Les serveurs de fichiers

De nombreuses machines fonctionnent comme des serveurs et présentent des informations qui peuvent être lues à distance. Ces machines peuvent être interrogées à volonté et l'on peut, sous certaines conditions, ramener sur sa propre machine les informations qui y sont rangées.

Les protocoles les plus courants

KERMIT
Le logiciel KERMIT est gratuit. Il est entretenu et distribué par des bénévoles. Il est originaire de l'Université de Columbia qui met à la disposition des utilisateurs plus de 100 versions différentes destinées à presque tous les ordinateurs existants. Les versions pour micro-ordinateurs permettent d'émuler un terminal et de transférer des fichiers selon un protocole très sûr mais relativement lent. Ce protocole est tombé en désuétude.

FTP (File Transfert Protocol) et NFS (Network File System)
Ces sigles désignent les protocoles de transfert spécialement orientés vers le transfert d'informations sur les réseaux fonctionnant selon le protocole TCP/IP.
Les informations sont débitées en paquets de petites tailles (≈1.5k) qui seront acheminés par le réseau selon les méthodes habituelles. A l'arrivée, les paquets sont regroupés pour recréer l'information initiale. Ceci évite que certaines machines n'accaparent les lignes de transmission pendant de trop longues périodes et permet d'entrelacer de nombreux transferts distincts sur les mêmes lignes.

SMB file transfer
SMB ("Server Message Bloc") est le protocole utilisé sur les réseaux Token-Ring d'IBM pour assurer les fonctions de serveurs et de transfert de fichiers. De nouveau, d'autres ordinateurs, disposant des connexions adéquates, peuvent utiliser ces protocoles.

AFP (Apple filling protocol)
Le protocole logique AFP (Apple Filing Protocol) définit les règles qui assurent les transferts de fichiers sur le réseau AppleTalk. Il est disponible sur tous les ordinateurs de la gamme Macintosh (et aussi, moyennant des interfaces convenables, sur la plupart des autres machines). Ce protocole est utilisé pour transmettre des informations aux imprimantes LaserWriter par exemple.

Netware de Novell
Il s'agit d'un serveur de fichiers qui était répandu dans l'environnement PC MS-DOS. Il utilise habituellement un réseau Ethernet comme support physique et permet le partage des fichiers situés sur un serveur entre plusieurs ordinateurs MS-DOS. Moyennant les logiciels adéquats, il est possible d'accéder au serveur à partir d'autres machines, comme un Macintosh (cf. NetWare for Macintosh).

Pour atteindre un serveur, il faut utiliser un programme spécialisé qui nécessite de remplir un certain nombre de paramètres qui sont habituellement les suivantes.

Hôte : l'adresse de la machine lointaine.
Identification : le nom d'utilisateur. Il est de tradition d'utiliser le nom "`anonymous`". Ceci signifie que l'on demande à utiliser les fonctions disponibles sans autorisation.
Mot de passe : lorsque le nom d'utilisation est "anonymous", il ne faut pas disposer d'un mot de passe (l'accès est libre), mais certaines machines, pour des raisons techniques, exigent quelque chose. De nouveau, il est de tradition d'utiliser sa propre adresse comme mot de passe (ex : zero.phe.ulg.ac.be). Si la case est vide, certains programmes fournissent un mot de passe par défaut.
Répertoire : c'est le nom du catalogue que l'on désire exploiter. En général, on peut laisser cette zone vide.

Dès que vous validez ce dialogue, le programme essaye d'entrer en contact avec la machine demandée. Dès que la liaison est établie, on voit apparaître une liste des fichiers disponibles sur le serveur. Un double clic sur l'un d'entre eux provoque son rappel sur votre machine.

☞ Les fichiers ainsi relus ne sont pas toujours directement utilisables. Certains d'entre eux sont "compactés" (.zip, .uu, .tar, etc...), ce qui signifie qu'ils ont été traités par un programme spécial qui en a réduit la taille (pour diminuer la durée du transfert). Il faut donc les décompacter à l'aide d'un logiciel adéquat. (Consulter par ex. http://www.stuffit.com/.)

3.3.- Le courrier électronique

Le courrier électronique permet l'échange d'informations entre différents utilisateurs sous forme de messages envoyés de personne à personne. Les transferts n'impliquent pas l'attention simultanée des deux partenaires. Un message électronique comporte toujours l'adresse du destinataire. Dès qu'on l'a déposé sur le réseau, un tissu mondial de machines spécialisées va le faire parvenir au destinataire. C'est souvent le réseau Internet qui assure le transport.

L'acheminement du courrier est très rapide (quelques dizaines de secondes) mais peut parfois prendre beaucoup plus longtemps (plusieurs jours en cas de difficultés).

Pour utiliser le courrier électronique, il faut disposer d'une "boîte aux lettres" électronique. Bien qu'il soit techniquement possible de placer cette boîte sur son micro-ordinateur personnel, il est recommandé de recourir à une machine spécialisée qui reste en marche 24 heures sur 24. A l'université, le SEGI ou les unités décentralisées assurent la gestion de quelques unes de ces machines sur lesquelles tous les membres du personnel et les étudiants peuvent obtenir une boîte aux lettres sur simple demande.

Le destinataire doit lire son courrier en "relevant" sa boîte. Cette opération se fait grâce à un programme qui va permettre aussi de gérer son courrier. On peut par exemple répondre à un correspondant, gérer une liste de contacts, faire des adressages multiples, etc…

Les avantages du courrier électronique sont nombreux. Il est moins "perturbant" (et moins cher) que le téléphone, plus rapide que la poste et permet de transmettre des documents "formatés" comme des textes compliqués (cf. le standard T_EX) ou des images.

Notons par exemple que de plus en plus d'éditeurs de revues scientifiques utilisent le courrier électronique pour échanger avec leurs auteurs les manuscrits et les épreuves d'imprimerie.

De nombreux programmes existent. Leur fonctionnement est simple et vous pouvez les utiliser pour envoyer des messages à vos professeurs par exemple.

Les noms de personne sur le réseau sont notés selon une syntaxe particulière :
un nom (qui identifie la personne), le signe @ (arobas) et le nom de la machine qui va recevoir le message. Cette machine doit être active en permanence et disposer d'un système de réception ad-hoc. Pour les physiciens de Liège, ce service est assuré par une machine du SEGI.

Exemples d'adresses locales :
`H.P. Garnir` hpgarnir@ulg.ac.be
`D. Strivay` dstrivay@ulg.ac.be

Pour envoyer un message, il suffit d'ouvrir votre gestionnaire de courrier et de choisir le menu Nouveau Message. Une fenêtre apparaît et il suffit d'y donner l'adresse du récipiendaire et le texte du message. Quand celui-ci est composé, le bouton "Envoyer" provoque l'envoi du message.

Un service webmail est aussi disponible en http://mail.ulg.ac.be

3.4.- Le world wide web (WWW)

Plusieurs programmes browser (ou butineur en français) sont disponibles, souvent gratuitement. Les pages contiennent des liens hypertextes qui sont présentés sous forme de textes soulignés ou d'icônes entourées d'une lisière bleue. Un clic sur ces éléments vous conduit directement sur une autre fenêtre qui correspond au nouveau document demandé.
L'URL du serveur de base de l'université est "www.ulg.ac.be".

4.- LES SERVICES AUXILIAIRES

4.1.- Le serveur PH

Une option utile de certains programmes (comme Eudora) permet de consulter un serveur PH ("phone book"). Ce serveur contient la liste de tous les membres de l'université et des étudiants qui en ont fait la demande. Le serveur PH de l'Université de Liège a comme addresse : `ph.ulg.ac.be.`
Une version interactive est accessible en http://annuaire.ulg.ac.be/

4.2.- Les name servers

A Liège, la gestion des names servers est assurée par le SEGI. Différentes machines sont utilisées pour ces activités. Pour la physique, ce sont les suivantes :
139.165.40.1 (machine AIX de l'unipc)
139.165.32.13 (Machine AIX du SEGI)

4.3.- TCP/IP

Pour qu'un ordinateur utilise les services de l'Internet, il faut qu'il dispose des programmes permettant d'utiliser le réseau TCP/IP. Ces programmes sont fournis par le constructeur comme des extensions au système.
Un utilitaire de configuration permet de choisir le mode de liaison (habituellement Ethernet ou une liaison modem en fonction des interfaces présentes) et précise les paramètres de connexion.

5.- LE CONCEPT DE DOCUMENTS "ACTIFS", LE LANGAGE "JAVA", php et xml

Les documents WWW contiennent habituellement des informations passives (textes et images) et des liens vers d'autres pages. Il est maintenant possible d'y inclure des pseudo-programmes (Applet) qui, lorsque le document est téléchargé, seront automatiquement exécutés localement sur la machine du client. Le document contient donc à la fois des informations et des méthodes (cf. objets). Le langage **JAVA** (proposé initialement par SUN mais maintenant repris par beaucoup d'autres sociétés) ainsi que le langage interactif **JavaScript** semblent être des candidats de choix pour cette nouvelle approche.
Les applets et JavaScript permettent à une page web d'être active sur votre machine. Une autre approche consiste à envoyer du serveur des pages générées dynamiquement en fonction des paramètres que vous avez transmis. Le code est exécuté sur le serveur. Ainsi le client ne reçoit que le résultat du script, sans aucun moyen d'avoir accès au code qui a produit ce résultat. C'est le principe de PHP. Vous pouvez configurer votre serveur web afin qu'il analyse tous vos fichiers HTML comme des fichiers PHP. Dans ce cas, il n'y a aucun moyen de distinguer les pages qui sont produites dynamiquement des pages statiques.

XML (Extensible Markup Language - en français : langage de marquage étendu) est une extension du langage html utilisé dans la construction des pages web.
Le langage html est constitué d'un certain nombre d'instructions (ou balises) qui, cachées dans le texte d'une page web, contrôlent la mise en page et le fonctionnement de la page. XML permet de construire dynamiquement de nouvelles balises et d'étendre ainsi la fonctionnalité des pages vers des domaines où les balises standard du html classique ne présenteraient pas assez de souplesse. On peut donc utiliser "un langage à l'intérieur d'un autre langage" et en quelque sorte télécharger en temps réel de nouvelles fonctionnalités. XML a permis de créer des sous langages html spécialisés comme SVG : Scalable Vector Graphics (module graphique), MathML (formulation mathématique), etc...

6.- L'INTERNET ET LES ÉTUDIANTS

6.1.- Modalités et déontologie d'accès Internet pour les étudiants ULg

L'Université de Liège offre à tous les étudiants réguliers en ordre de paiement d'inscription la possibilité d'accéder gratuitement à l'infrastructure réseau de l'ULg ("ULgNet") et, par cette voie, au réseau informatique mondial INTERNET. Cette possibilité n'est cependant offerte à l'étudiant qu'après son engagement à respecter la déontologie d'usage du réseau.

L'étudiant reçoit un identifiant personnel (composé de la lettre "s", en minuscule, suivi de son numéro de matricule), auquel est associé un mot de passe personnel, indiqué en annexe de la carte d'étudiant si la demande est effectuée lors de l'inscription ou mentionné sur un document spécifique si la demande est effectuée postérieurement à l'inscription. L'identifiant et le mot de passe peuvent être utilisés 24 heures après leur communication à l'étudiant.
Du fait de cette demande, l'étudiant accepte que figurent dans l'annuaire électronique consultable par Internet ses nom, prénom, section et, lorsqu'il en possède une, son adresse de courrier électronique.
L'identifiant et le mot de passe personnels constituent ensemble la clé d'accès aux services Internet offerts par l'ULg à ses étudiants.

L'usage de l'Internet suppose de la part de l'étudiant une maîtrise suffisante des outils informatiques d'exploitation de ce réseau et aucune information n'est dispensée, autre que :
• celle qui peut être acquise dans le cadre des cours ;
• celle qui est disponible dans les documents en format électronique sur le réseau (informations et instructions relatives aux programmes de courrier électronique ou aux programmes de communication) ;
• celle qui peut être acquise au travers des échanges d'expérience entre étudiants.

Du simple fait de sa demande d'accès au réseau, l'étudiant est soumis à la réglementation d'usage de l'Internet et particulièrement de sa composante belge réservée à l'enseignement et à la recherche : l'infrastructure fédérale BELNET.

En particulier, il est interdit d'utiliser le réseau pour toute activité ayant pour résultat :
• d'accéder de façon illicite aux ressources des réseaux connectés ;
• de nuire au fonctionnement du réseau ou de mettre en péril l'utilisation ou les performances du service pour les autres utilisateurs ;
• de dépenser inutilement des ressources (hommes, capacité du réseau, ordinateurs) ;

• de compromettre la vie privée des utilisateurs ;
• une exploitation commerciale du réseau.

Le droit d'accès est strictement personnel, il est interdit de le communiquer à quiconque.

Toute diffusion du mot de passe (notamment à des fins commerciales), qui aurait comme résultat une exploitation abusive du réseau, expose le détenteur à des mesures à la discrétion de l'ULg, sans préjudice des actions que pourrait intenter toute partie s'estimant lésée par cette diffusion.

6.2.- L'identifiant et le mot de passe ULg pour les étudiants

Pour les étudiants, l'identifiant ULg a la forme d'une chaîne de caractères commençant par la lettre "s" minuscule suivie du numéro de matricule étudiant. Grâce à cet identifiant et ce mot de passe, l'utilisateur pourra demander (de manière automatisée via un serveur) la création d'une boîte aux lettres électronique (Email) ou l'autorisation d'accéder par téléphone au réseau depuis le domicile, ou encore d'autres services qui seront offerts ultérieurement. Enfin, l'identifiant et le mot de passe sont utiles dans l'utilisation courante de ces services (à chaque fois qu'une validation d'identité est nécessaire).

6.3.- Description générale du service d'Email offert par le SEGI aux étudiants

L'ouverture d'une boîte aux lettres électronique par un étudiant qui dispose déjà de son identifiant et mot de passe ULg, nécessite une activation telle que mentionnée dans le document "modalités et déontologie d'accès Internet pour les étudiants ULg" remis à l'étudiant lors de son inscription.

Par cette activation, l'étudiant rend opérationnelle son adresse électronique, qui sera toujours de la forme normalisée :

identifiant@student.ulg.ac.be

où identifiant est constitué du numéro de matricule préfixé par la lettre "s" (minuscule) ; cette adresse correspond à une boîte aux lettres électronique sur un "serveur de courrier" de l'ULg. Le courrier à destination de l'étudiant est déposé dans sa boîte aux lettres, et y reste jusqu'à sa relève ou, à défaut, pendant 2 semaines.

Notez que l'ouverture d'une boîte aux lettres se traduit par l'insertion automatique de l'adresse Email associée dans le répertoire électronique de l'ULg, aux côtés de vos coordonnées personnelles.

Le bon fonctionnement du système de courrier électronique n'est garanti que si le logiciel d'Email utilisé est correctement configuré. Les paramètres de configuration indispensables sont personnels ; il ne faut donc pas utiliser ceux d'un autre utilisateur. Ces paramètres sont affichés au moment de l'activation de la boîte aux lettres et peuvent être réaffichés à tout moment par après.

6.4.- Autres sujets traités (cf. http://www.ulg.ac.be/segi/student/email/)

• Obtenir une boîte aux lettres électronique personnelle.
• Faire suivre le courrier de son adresse normalisée ULg vers une autre adresse.
• Désactiver le renvoi du courrier vers une autre adresse.
• Réafficher les paramètres de sa boîte aux lettres électronique personnelle.
• Tester la configuration de son logiciel d'Email.

C H A P I T R E IX

Le graphisme

1.- LE GRAPHISME

La programmation des extensions graphiques est souvent délicate car il n'existe pas encore de norme bien définie en ce qui concerne le graphisme. Chaque constructeur fournit directement ou sous forme d'options des instructions graphiques spécifiques qui permettent de créer des dessins en utilisant au mieux les ressources graphiques de sa machine.

Actuellement, tous les ordinateurs disposent d'un écran graphique et tout ce qui apparaît à l'écran doit être considéré comme du dessin. La notion d'écran textuel elle-même est simulée (cf. la fenêtre de la console). Le graphisme repose sur des gestionnaires spécialisés qui isolent l'utilisateur du matériel en lui fournissant un environnement graphique identique sur toutes les machines d'une même gamme.

Ces systèmes reposent tous sur la technologie des écrans à haute résolution.

2.- LE MODE HAUTE RESOLUTION "BIT-MAP" OU "PIXEL MAP"

Le mode haute résolution travaille habituellement sur les micro-ordinateurs selon le principe de l'écran "bit-map" (carte de bits). Chaque point élémentaire de l'écran (appelé "pixel" pour PIcture X ELement) peut être allumé ou éteint par le programme. Si l'affichage est noir/blanc, un pixel correspond à un seul bit. En couleurs, chaque pixel correspond à un certain nombre de bits (de 2 à 48) dont chaque état correspond à une couleur. Ces informations sont rangées en mémoire. En affichage graphique (mode "raster"), on visualise à l'écran le contenu d'une zone particulière de la mémoire réservée spécifiquement à cet usage.
Les modes graphiques couvrent au moins le mode VGA qui offre une résolution de 640 points horizontaux sur 480 points verticaux, soit une grille de 307200 pixels pouvant apparaître chacune en 16 teintes différentes. Le bit-map occupe alors ≈160k de mémoire. L'augmentation de la taille des écrans et des performances des machines permettent de gérer des écrans graphiques présentant des nombres de pixels de plus en plus grand. La dimensions minimale la plus courantes est maintenant de 1024 x 768, mais des écrans plus grands sont souvent proposés.

3.- LES TECHNIQUES DE DESSIN

Il existe différentes techniques qui permettent d'utiliser un écran graphique. Nous envisagerons les techniques cartésienne, tortue, bloc et table de formes.

3.1.- La technique "cartésienne" ou xy

Elle consiste à voyager sur l'écran d'un point à un autre en donnant les coordonnées XY du point de départ et du point d'arrivée repérés dans un système d'axes fixes. En pratique, le point 0,0 se trouve en haut à gauche de l'écran et les coordonnées d'un point varient entre 0..639 pour X et 0..479 pour Y.

3.2.- La technique "tortue"

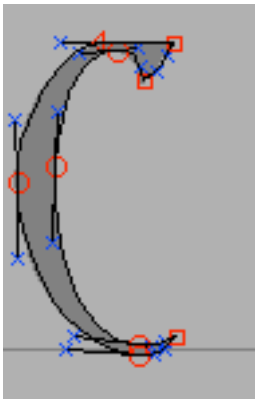
La tortue symbolise un instrument qui permet de voyager sur l'écran en donnant un angle et un déplacement. La tortue tourne de l'angle demandé et avance de la longueur du déplacement. Le parcours est donc décrit selon une suite de rotations et de déplacements et non plus en terme de coordonnées. (C'est le mode utilisé en LOGO par exemple.)

3.3.- La technique par "bloc"

La technique par "bloc" consiste à reproduire, dans des variables de structure adéquate, des portions d'images de type "bit-map" et ensuite de les recopier rapidement dans la zone graphique. La zone recopiée est souvent un rectangle et la variable, une matrice rectangulaire. Remarquons que cette technique est utilisée par la plupart des micro-ordinateurs pour afficher du texte. Les images des lettres sont rangées dans des matrices de différentes tailles et sont recopiées à partir d'une zone de mémoire morte qui sert de "générateur de caractères".

3.4.- La technique par table de formes

La technique par "table de formes" contient une liste de "vecteurs de tracés", groupés selon une syntaxe complexe, qui permettent de tracer des formes préétablies. Cette technique s'apparente à la programmation, et l'on parle habituellement de langage de description de page (PDL - Page Description Langage) pour définir les primitives et la syntaxe. Cette technique est utilisée pour décrire la forme des lettres et des dessins reproduits sur certaines imprimantes de haute qualité. Les graphiques sont donnés sous forme d'une table de formes contenant les coordonnées de chacun des points décrivant le symbole à reproduire. Le logiciel associé permet de changer la taille, l'inclinaison et la position du symbole tout en optimisant la qualité du dessin en fonction des performances de l'imprimante ou de l'écran (cf. les polices Adobe et la technique TrueType).



Exemple d'une lettre (ici c) définie par une table de points.

4.- LES COULEURS

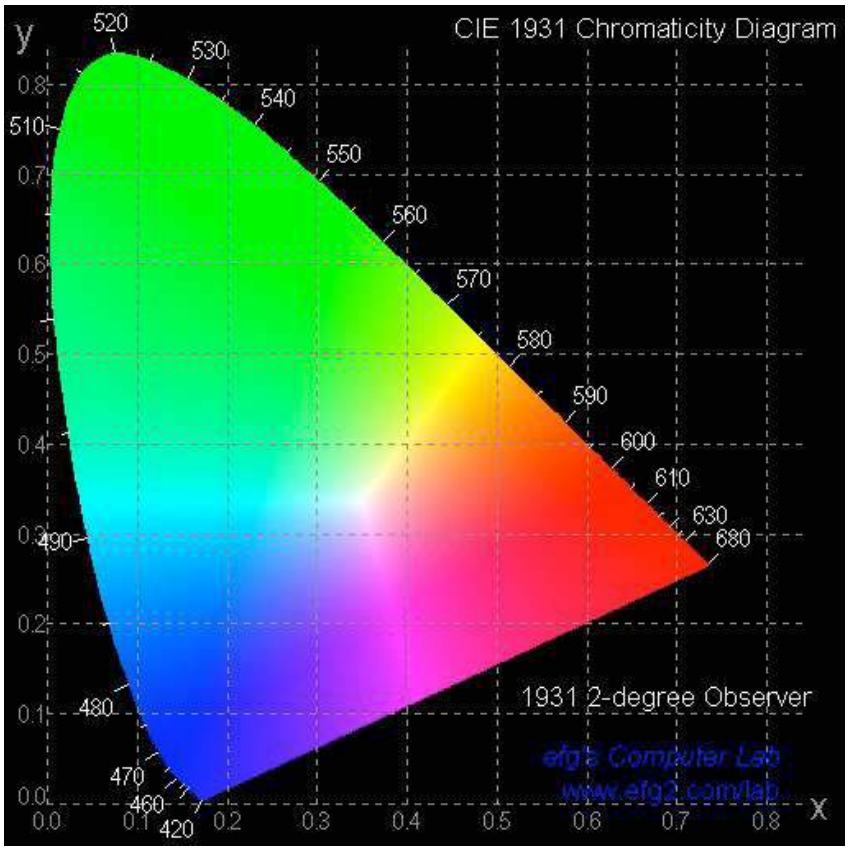
Les écrans couleurs des ordinateurs fonctionnent selon les mêmes principes que la télévision couleurs. La génération des teintes est basée sur la synthèse trichromatique RVB (rouge, vert, bleu).

L'oeil analyse la lumière en fonction de son intensité et de sa longueur d'onde (ou de sa fréquence). Cependant, le mécanisme physiologique de perception des couleurs ne se base pas sur une analyse spectrale du signal lumineux. L'oeil intègre les différentes longueurs d'onde et les perçoit toujours comme une seule couleur. Une lumière formée par un mélange de diverses longueurs d'onde sera vue comme une seule teinte, l'oeil étant incapable de séparer les différentes composantes d'une lumière colorée (ex. : jaune+bleu = vert ...). Toute la technologie de la reproduction en couleurs est bâtie sur cette constatation.

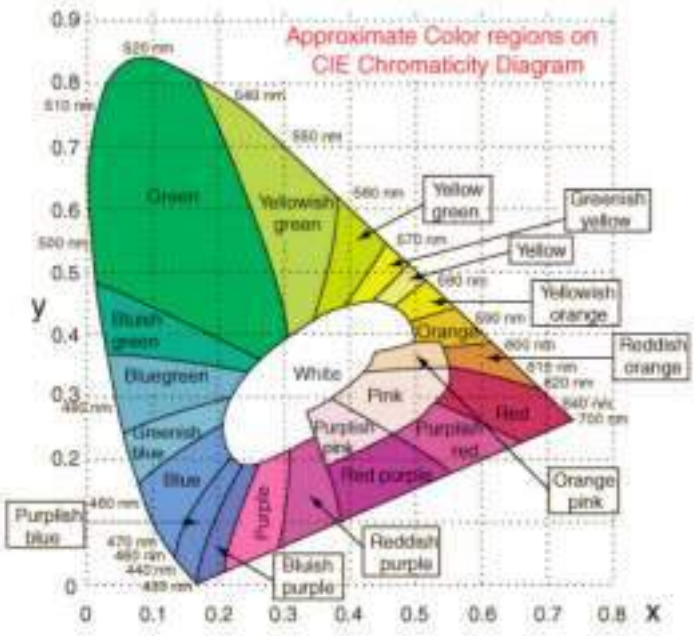
L'étude expérimentale de la sensibilité de l'oeil aux couleurs a montré qu'il est possible de simuler presque toutes les teintes en mélangeant seulement **trois** longueurs d'onde habilement choisies (synthèse trichromatique : lois de Grassmann). Chacune des teintes perceptibles par l'oeil peut donc être décrite par trois paramètres qui correspondent aux trois intensités des couleurs fondamentales qu'il faut utiliser pour la reconstituer.

Si l'on normalise à une intensité totale constante, la somme des trois paramètres reste égale, ce qui réduit à deux le nombre de paramètres libres et permet une représentation graphique plane. Depuis 1931, on utilise le graphique proposé par la C.I.E. (Commission Internationale de l'Eclairage) dans lequel la région des couleurs perçues par l'oeil est représentée par une surface comprise dans le triangle formé par les axes X et Y et la droite X+Y=1. (Notons que les trois couleurs fondamentales utilisées dans cette représentation sont “irréelles” car situées hors de la zone de perception de l'oeil...)

A partir d'un tel diagramme, chaque teinte est définie par les deux coordonnées X et Y qui mesurent l'effet physiologique de la lumière et non sa longueur d'onde. Dans l'industrie, ces paramètres sont couramment utilisés pour caractériser les verres teintés (feux de signalisation) ou les colorants.



Graphique C.I.E. permettant une classification systématique des teintes colorées
cf. par ex. http://www.cs.rit.edu/~ncs/color/a_chroma.html
http://www.efg2.com/Lab/Graphics/Colors/Chromaticity.htm



http://www.mat.univie.ac.at/~kriegl/Skripten/CG/node9.html

5.- REPRESENTATION NUMERIQUE DES IMAGES

5.1.- Image bit-map (peinture - paint)

Il s'agit d'un mode graphique qui s'apparente à la peinture ou, plus exactement, à une mosaïque formée d'une multitude de petites tuiles colorées. (Ces tuiles sont appelées "Pixels" pour Picture Elements.) Les lignes sont décrites comme des rangées de pixels adjacentes et toutes les images sont définies uniquement par un assemblage de tuiles. Comme dans les vraies mosaïques, on peut changer une image "bit-map" en changeant la couleur des tuiles. On peut produire des teintes inexistantes en juxtaposant habilement des tuiles de différentes couleurs pour donner à distance l'illusion de teintes intermédiaires ("dithering"). Ceci permet, entre autres, de simuler des grisés lorsqu'on ne dispose que de tuiles noires et blanches.

Les images noires et blanches n'ont besoin que d'un seul bit de donnée pour décrire l'état de chaque pixel. Pour les images en couleurs, chaque pixel doit être représenté par plusieurs bits. La résolution de 8 bits/pixel permet 256 teintes différentes. Pour obtenir une image de qualité photographique, on peut aller jusqu'à 24 bits par pixels (16.8 millions de teintes possibles). Une grande image bit-map de haute résolution peut occuper une importante zone de mémoire ou d'espace disque!

La structure en mosaïque des images bit-map rend leur manipulation assez délicate. Pour faire tourner, agrandir ou déplacer une zone, il faut agir sur chacune des tuiles individuellement et effectuer un très grand nombre d'opérations délicates. De plus, lorsque l'on superpose une image sur une autre, les tuiles originales sont remplacées par la nouvelle image et l'image sous-jacente est définitivement perdue. Lorsqu'on agrandit ou modifie la densité des tuiles d'une image bit-map, on produit généralement une nouvelle image qui présente des contours en dents de scie et les nuances de teintes produites par "dithering" sont changées de façon irréversible.

Néanmoins les images bit-map permettent d'obtenir des effets artistiques s'apparentant à ceux produits par la peinture traditionnelle. Notons aussi que les analyseurs d'images (scanner) et les programmes de retouche photographique travaillent toujours sur des images bit-map.

Lors de l'impression sur des imprimantes de qualité, le résultat est souvent décevant car les images bit-map ne permettent pas facilement de profiter de la plus grande résolution de ces imprimantes.

5.2.- Image objet (dessin - drawing)

Ces images sont composées d'objets décrits mathématiquement sous forme d'une suite de vecteurs de tracé ou de régions polygonales de différentes couleurs. La mémorisation de ces images se résume à une suite d'instructions graphiques décrivant les opérations élémentaires à réaliser pour reproduire le dessin. De plus, les teintes et les couleurs des régions peuvent être définies avec précision même si, à l'écran, ces nuances ne sont pas reproductibles.

Les programmes de gestion de ces images objets maintiennent une base de données contenant les éléments graphiques composant l'image et mémorisent les posi-

tions relatives et les teintes de tous les objets. Il est donc possible de superposer deux objets sans perdre l'information décrivant l'objet masqué. Les opérations de déplacement, rotation, changement de taille, etc… se résument à des opérations mathématiques sur les objets et n'induisent pas de dégradation appréciable lorsqu'on les effectuent. Et comme la plupart de ces opérations sont réversibles, on peut toujours revenir à la situation initiale sans perte d'information… Ces programmes présentent habituellement à l'utilisateur un dessin approché (tenant compte de la résolution limitée de l'organe de présentation) qui permet de contrôler l'image et de changer interactivement les caractéristiques des objets. L'opérateur visualise immédiatement l'effet de ses manipulations mais la résolution limitée des écrans ne permet pas toujours de juger exactement du résultat.
Les images objets sont spécialement adaptées à la réalisation de documents techniques comme des plans, des graphiques, etc… car elles ne sont pas tributaires de la résolution des périphériques de visualisation et d'impression. En effet, lors de l'impression, les images des objets graphiques sont calculées avec grande précision par des méthodes mathématiques rigoureuses et représentées sous un aspect aussi régulier que la qualité de l'imprimante le permet.

5.3.- Formats techniques de représentation des images

Il n'existe pas, à l'heure actuelle, de standard décrivant le contenu des fichiers images. Il y a cependant certains formats qui sont plus utilisés que d'autres. Nous en évoquons quelques-uns ci-dessous.

5.3.1.- TIFF (Tag Image File Format)

Ce format a été créé pour mémoriser les images bit-map produites par les scanners. Il existe différentes versions qui diffèrent par le nombre de teintes : Monochrome TIFF, Gray-Scale TIFF et Color TIFF. Attention, le format TIFF est sujet à certaines nuances qui compliquent parfois le transfert d'images d'un programme à un autre.

5.3.2.- PostScript

PostScript n'est pas à proprement parler un format mais plutôt un langage de description de page (PDL — Page Description Language). Il fut utilisé pour la première fois en 1986 dans l'imprimante laser "LaserWriter" d'Apple. La firme Adobe, qui a développé ce langage, a ensuite accordé des licences d'utilisation à plusieurs autres constructeurs d'imprimantes et petit à petit le langage est devenu un standard dans le domaine de la micro édition. Un fichier PostScript est un fichier de texte qui contient le programme (en langage PostScript) qui permet de décrire une image ou un texte formaté.
Ce format est utilisé pour transmettre les informations graphiques vers les imprimantes laser qui acceptent le standard PostScript.

5.3.3.- EPS (Encapsulated PostScript)

Les fichiers EPS contiennent généralement deux versions du dessin : une version textuelle qui décrit le dessin en langage PostScript et une seconde version qui contient une image bit-map à faible résolution du même dessin. Cette image est destinée à faciliter le contrôle et le positionnement de l'image dans les programmes de mise en page électronique (comme PageMaker). A l'impression, le fichier PostScript (qui est généralement très complexe) remplace l'image approximative et donne un résultat de haute qualité.

5.3.4.- PICT, PICT2, PICS

C'est le format utilisé par l'ordinateur Macintosh d'Apple pour définir ses graphiques. Les fichiers PICT énoncent (sous une forme synthétique) la suite des instructions QUICKDRAW qui permettent de reproduire un dessin à l'écran ou sur imprimante. Le format PICT original est limité à 8 couleurs. PICT2 étend le format à un nombre illimité de couleurs et correspond aux instructions de Color Quickdraw. Le format PICS est un format qui permet de mémoriser plusieurs images PICT (ou PICT2) et de les reproduire en séquence pour simuler un effet de dessin animé.

5.3.5.- GIF

La norme GIF (Graphic Interchange Format – G87a et G89a) définit un mode de codage qui réduit fortement la taille des fichiers sans perte de qualité (le codage est réversible). GIF convient bien aux images qui contiennent des zones de couleur uniforme (comme par exemple les images de bandes dessinées ou les petits logos). GIF est limité à 256 couleurs (8 bits/pixel) et accepte que l'une des "couleurs" soit transparentes. C'est le format le plus utilisé sur le WWW pour transmettre les petites images et logos.

La norme GIF89a permet deux choses supplémentaires.

1.- Coder les images sous format entrelacé. La première partie du fichier décrit l'image très grossièrement, les détails de plus en plus précis sont décodés ensuite. Ces images apparaissent rapidement sous une version "floue" avant de devenir de plus en plus nettes. Cette technique est habituelle sur le WWW.

2.- Enchaîner des images GIF de même taille qui apparaîtront sous forme d'une séquence animée à un rythme convenu. (Pour construire ces séquences, il faut un utilitaire spécialisé.)

Notons que le format GIF ne convient pas aux images de type "photographique" contenant beaucoup de détails et de couleurs.

5.3.6.- JPEG

La norme JPEG (Joint Photographic Expert Group) définit un mode de compression qui permet de réduire fortement la taille d'un fichier image. JPEG accepte cependant un certain degré de "perte" lors de la compression, ce qui signifie que l'image recomposée est légèrement moins précise que l'image originale (le codage est non réversible mais, pour l'oeil humain, la différence est imperceptible). JPEG est spécialement adapté aux images du type photographique possédant une gamme importante de couleurs. Des paramètres permettent de jouer sur les facteurs vitesse/fidélité/niveau/décompression. Selon les réglages, la réduction de taille des fichiers peut atteindre un facteur 100.

Notons que JPEG conserve la palette de couleurs (max. 16 millions) mais ne convient pas aux documents noir/blanc ou aux dessins au trait.

5.3.7.- PNG

La norme PNG (Portable Network Graphic) remplace progressivement le format GIF et est de plus en plus utilisé car il n'y a pas de "droit d'auteur" à payer pour l'utiliser. Il s'agit d'un mode de compression réversible (sans perte) qui accepte des images avec un résolution de 24 à 48 bits couleurs plus un canal alpha (transparence) et qui encode aussi les réglages d'affichage (gamma et commentaires) et les palettes de couleurs. Notons cependant que PNG ne supporte pas les GIF animés. (cf. http://www.libpng.org/)

5.3.7.- Format cinéma (movies)

• MPEG (Moving Pictures Experts Group)

Il s'agit d'une norme de compression spécialement orientée vers la transmission d'images animées. La norme n'est pas encore complètement définie (996 MPEG-4) mais elle permet de standardiser l'archivage et la visualisation de séquences animées. L'encodage, qui est irréversible (perte de qua-

lité), dépend d'un paramètre de qualité qui permet de comprimer plus ou moins fortement un document pour le rendre par exemple compatible avec la taille d'un DVD classique (4.7G). Les niveaux de qualité souvent utilisés sont High (1h/DVD), Standard (super 2h/DVD), LP, Long Play (3h/DVD), EP (extended play 4h/DVD).
Cette norme est aussi adoptée pour la télévision digitale à haute résolution.

• QUICKTIME et QUICKTIME VR (.MOV)
C'est une norme introduite par Apple en 1990. Elle permet la visualisation à l'écran de séquences animées. Il existe des "drivers" à la fois pour les ordinateurs Apple (qui contiennent le lecteur en standard) et pour les machines MS-DOS. L''option VR ajoute la réalité virtuelle au système, c'est-à-dire la possibilité de faire bouger l'image interactivement.

• AVI (Audio Video Interleave- audio vidéo entrelacée)
Cette norme, introduite par Microsoft en 1998 permet de stocker dans un même fichier des images vidéos et des sons (sous divers formats) (visualisation de films à l'écran avec son en plusieurs langues par ex). Il existe des "drivers" à la fois pour les ordinateurs Windows, Apple et Linux.

• DivX®
C'est un système CODEC (COdeur/DECcodeur) introduite en 2000 qui permet de comprimer très efficacement les films en vue de les faire circuler par exemple sur Internet (c'est l'équivalent du mp3 pour la vidéo). Il s'agit d'un produit commercial qui est actuellement disponible quasi gratuitement (mais qui pourrait ne pas le rester!).

• Realplayer (.ra), mp3, etc...
Dans la forêt des CODEC pour les sons et les images, on trouve tout un arsenal de formats plus ou moins répandus ayant chacun des caractéristiques plus ou moins spécifiques. Les compagnies privées qui distribuent des documents recherchent par exemple des formats "incopiables" qui, en théorie, limiteraient le piratage. Il est probable que dans un avenir proche, seuls quelques formats subsisteront.

C H A P I T R E X

Quelques algorithmes fondamentaux

1.- INTRODUCTION

La résolution par ordinateur d'un problème numérique revient généralement à définir l'algorithme, qui décrit de façon systématique la marche à suivre pour obtenir la solution, et ensuite à vérifier la qualité de cet algorithme en le codant dans un langage approprié. De nombreux travaux ont été consacrés à l'élaboration d'algorithmes performants destinés aux ordinateurs. Il s'agit d'un sujet très vaste qui peut faire l'objet d'études approfondies dans les cours traitant d'algorithmique ou d'analyse numérique.

Nous allons nous limiter à énumérer quelques “recettes” de calcul qui permettent de résoudre des problèmes classiques souvent rencontrés dans la pratique. Les solutions proposées ont été choisies en tenant compte à la fois de leur efficacité et de leur simplicité. Elles ne sont pas toujours les meilleures et nous engageons vivement le lecteur à consulter les nombreux ouvrages qui traitent des méthodes de calculs numériques lorsqu'il ne sera pas satisfait par les solutions proposées.

Les exemples de programmes sont donnés en langage Pascal et en JavaScript. Le JavaScript est disponible sur tous les “Browsers” modernes. Une fenêtre interactive permettant de faire fonctionner ces programmes est disponible en :

http://www.ulg.ac.be/ipne/etudiants/JAVASCRIPTMS.html (écran 17”) et

http://www.ulg.ac.be/ipne/etudiants/JSpresentation.html (écran VGA).

2.- NOTES SUR LA PRECISION DES CALCULS

Il faut garder à l'esprit que la précision des calculs réalisés par un ordinateur n'est pas infinie. Ceci est dû à des contraintes techniques qui limitent le contenu d'une variable à un nombre fini d'états différents alors qu'il existe une infinité de nombres. La machine doit donc choisir la représentation interne qui se rapproche le plus de la valeur demandée. L'erreur entre ces deux valeurs est l'“erreur d'arrondi”. Elle est généralement de l'ordre de 10^{-9} à 10^{-15}, mais lorsqu'elle est amplifiée par des méthodes de calcul peu délicates, elle peut conduire à des résultats complètement faux. La plupart des traités d'analyse numérique étudient en détail le problème des erreurs, nous renvoyons donc à ces ouvrages le lecteur intéressé. Signalons cependant quelques opérations à éviter.

• Calculer la différence de deux nombres quasi égaux :
`(1 E25 +1) - 1E25` ne donne pas 1!

• Additionner un grand nombre de fois un petit incrément :
`s:=0; for i:=1 to 1000 do s:=s+1/3` ne rend pas exactement 1000/3!

• Comparer directement deux nombres réels :
`if a=b then …` : a=b n'est vrai que si a est exactement égal à b. Il vaut mieux utiliser l'expression suivante lorsqu'on veut comparer des nombres réels :
`if abs(a-b)<epsilon then …`
où epsilon est une constante légèrement supérieure à l'erreur d'arrondi de la machine utilisée.

3.- QUELQUES ALGORITHMES FONDAMENTAUX

3.1.- Evaluation de la somme des éléments d'un tableau

Pour calculer la somme des éléments d'un tableau d'une façon élégante, on utilise une boucle qui additionne un à un les éléments du tableau à une variable préalablement mise à zéro.
Soit s la variable qui va contenir la somme des a[1..n]. On fait

```
s=0;
for(i=1;i<n;i++) s=s+a[i];
```

3.2.- Permutation du contenu de deux variables

Pour permuter le contenu des deux variables A et B, on ne peut pas utiliser le code suivant : `A=B; B=A;` car la première affectation remplace le contenu de A par celui de B, ce qui efface le contenu original de A qui n'est plus disponible dans la seconde instruction!
Pour réaliser correctement l'opération, il faut nécessairement utiliser une variable tampon C qui préservera la valeur de A, soit

```
C:=A;
A:=B;
B:=C;
```

3.3.- Evaluation d'un polynôme : Règle de Horner

L'évaluation numérique de la valeur d'un polynôme peut se faire en alternant les multiplications et les additions.

Exemples :
Le polynôme $P(x) = ax^3 + bx^2 + cx + d$ sera codé

```
P:=((a*x+b)*x+c)*x+d;
```

Lorsque les coefficients du polynôme sont rangés dans un vecteur a[0..n], on peut évaluer numériquement $P(x) = \sum a[i]\, x^i$ par les instructions suivantes, en Java :

```
p=0;
for(i=n;i>=0;i--) p=p*x+a[i];
```

Soit à évaluer

$12x^4+6x^3-2x^2+7x+3 \rightarrow ((((12x)+6)x-2)x+7)+3$

ce qui donne en JavaScript

```
coef= new Array(12,6,-2,7,3);
x=prompt("x=","0");
p=0;for (i=0;i<5;i++)p=p*x+coef[i];
document.write("En "+x+" le poly. vaut "+p);
```

3.4.- Interpolation polynomiale entre n points : Formule de Lagrange

Lorsqu'on dispose de n couples de valeurs (x_i,y_i), le polynôme de degré n-1 tel que $p(x_i)=y_i$ est donné par

$$p(x) = \sum_{j=1}^{n} y_j \prod_{i \neq j, i=1}^{n} \frac{x - x_j}{x_j - x_i}$$

Cette expression est souvent utilisée pour construire une fonction continue à partir de valeurs numériques tabulées pour certaines valeurs de x. (Notons que la formule se simplifie si les x_i sont équidistants.)

Elle peut aussi servir à retrouver les N coefficients qui permettront d'exprimer le polynôme sous sa forme classique $P(x) = \sum a[i]\, x^i$, plus simple à calculer. Notons que l'ajustement par moindres carrés d'un polynôme paramétrique de degré N-1 conduit (en théorie) au même résultat.

3.5.- Système d'équations linéaires à coefficients constants : Méthode de Gauß

La méthode d'élimination de Gauß permet de résoudre numériquement un système d'équations linéaires à coefficients constants (pour autant que le système ne soit pas singulier et que sa dimension ne soit pas trop importante). Elle est basée sur la propriété des systèmes linéaires qui nous apprend que l'on peut ajouter à n'importe quelle équation une combinaison linéaire des autres sans changer la solution. La recherche de la solution consiste à transformer le système en un système plus simple (mais équivalent) en annulant le plus de coefficients possibles.

Un exemple montrera le cheminement du processus d'élimination. Soit le système de 3 équations :

```
   x +  y - 2z = 2
   x + 3y - 4z = 8
 -2x - 4y +10z =-2
```

Soustrayons la première équation de la deuxième, le système devient :

```
   x +  y - 2z = 2
   0 + 2y - 2z = 6
 -2x - 4y +10z =-2
```

Ajoutons 2 fois la première équation à la troisième, le système devient :

```
   x +  y - 2z = 2
   0 + 2y - 2z = 6
   0 - 2y + 6z = 2
```

Ajoutons une fois la deuxième équation à la troisième, le système devient :

```
   x +  y - 2z = 2
   0 + 2y - 2z = 6
   0 +  0 + 4z = 8
```

La troisième équation donne z=2. En remplaçant z par sa valeur dans la deuxième, on trouve y=5 et la première équation donne après remplacement x=1, CQFD.

Le processus d'élimination consiste à construire un système triangulaire qui donne directement la valeur d'une des racines. A partir de cette racine, les autres équations permettent de retrouver successivement toutes les solutions. Le codage automatique de cet algorithme est relativement simple et peut être trouvé dans de nombreux livres traitant de la programmation (cf. le programme GAUSS.PAS par exemple…).
La plupart des codages utilisent des astuces pour réduire au minimum les erreurs d'arrondis dans les calculs. Néanmoins, celles-ci finissent par se cumuler, ce qui limite la précision des résultats dès que la taille du système dépasse 5 à 10 équations. (On peut estimer qu'il faut 2 chiffres significatifs par équation donc les 12 chiffres significatifs des nombres de type “double” permettent au maximum environ 6 équations.) Notons qu'il est toujours possible de vérifier a posteriori l'exactitude de la solution en réinjectant la solution dans le système.

3.6.- Intégration numérique : Méthode de Simpson

L'intégration numérique d'une fonction consiste à estimer la surface sous la courbe entre les bornes d'intégration. La méthode la plus simple est celle dite "du rectangle" qui calcule la somme des surfaces de rectangles étroits ayant comme hauteur une valeur de la fonction. La règle de Simpson, basée sur le même principe, est plus précise et à peine plus compliquée, elle est recommandée en pratique. La voici sous une forme adaptée au calcul.

$$\int_a^b f(x)dx = \sum_{i=0}^{n-1} \frac{x_{i+1} - x_i}{6} \left[f(x_i) + 4f(\frac{x_{i+1} + x_i}{2}) + f(x_{i+1}) \right]$$

où les $x_0..x_n$ sont n+1 valeurs de x réparties uniformément entre a et b.

Un codage en Java serait par exemple :

```
public class Integration {

    public static class MaFonction{
```

```
        public double f(double x){
            //la fonction à intégrer - ici:  3 x^2 + 4 x - 100
            double y;
            y= (3*x+4)*x -100;
            return y;
          }
    }

    public static double intSimpson(double a,double b,int n, MaFonction o){
        int i,z;
        double s,w;
        n=n+n;
        s=o.f(a)+o.f(b);
        w=(b-a)/n;
        z=4;
        for(i=1;i<n;i++){
            s=s+z*o.f(a+i*w);
            z=6-z; //z passe alternativement de 4 à 2
          }
        s=(s*w)/3.0;
        return s;
    }

    public static void main (String args[]) {
        System.out.println(intSimpson(0,10,100,new MaFonction()));
    }
}
```

3.7.- Méthode de classement : a - "Bubble sort"

Une méthode de classement très simple consiste à comparer deux éléments successifs d'une liste et de les permuter s'ils ne sont pas correctement ordonnés. Cette comparaison est effectuée successivement pour tous les éléments de la liste et est reprise jusqu'à ce qu'on n'ait plus de permutation à effectuer. Le fragment de programme suivant classe en ordre ascendant les éléments du vecteur a[n].

En Java

```
do{
 ok=true;          {a priori la liste est ordonnée}
 for(i=1;i<n;i++){
    if (a[i-1]>a[i]) {
     tampon:=a[i-1];
     a[i-1]:=a[i];
     a[i]:=tampon;
     ok=false;    {signale qu'une permutation au moins a eu lieu…}
while (!ok);
```

En JavaScript

```
function bubble(t){
 do{
  fini=true;
  for(i=1;i<t.length;i++)
     if(t[i-1]>t[i]){T=t[i-1];t[i-1]=t[i];t[i]=T;fini=false;}
  }while (!fini);}
function montre(t){
 for(i=0;i<t.length;i++)document.write(i+1+" | "+t[i]+"<br>")
 document.write("<hr>")}
dim=7;
tableau=new Array;
for(i=0;i<dim;i++)tableau[i]=Math.round(Math.random()*100);
montre(tableau);
bubble(tableau);
montre(tableau);

//-----variante avec chronomètre - vérifier la loi en n2

dim=prompt("Nombre d'éléments",100);
tableau=new Array;
for(i=0;i<dim;i++)tableau[i]=Math.round(Math.random()*100);
timea=new Date();
bubble(tableau);
timeb=new Date();
document.write("Pour "+dim+" éléments : "+(timeb-timea)+'ms');
```

Cette méthode est conceptuellement très simple mais très peu efficace. Elle est à proscrire énergiquement si les éléments à classer sont en nombre important et s'ils ne sont pas déjà presque ordonnés (car le temps d'exécution augmente comme N*N).
La méthode suivante, bien que plus compliquée à comprendre, est optimale dès qu'il s'agit de classer un grand nombre d'éléments.

3.8.- Méthode de classement : b - "Quick sort"

Cette méthode de classement a été inventée par C.A.R. Hoare en 1960. Il s'agit d'une méthode récursive conceptuellement complexe qui consiste à

1.- choisir un élément de la liste (le pivot),

2.- séparer la liste en deux sous-listes, l'une étant composée des éléments supérieurs ou égaux au pivot, l'autre des éléments inférieurs,

3.- reprendre successivement le raisonnement sur chacune des deux listes ainsi formées jusqu'à ce que les sous-listes se réduisent chacune à un seul élément. La procédure suivante réalise ces opérations (noter qu'il s'agit d'une procédure récursive et que les sous-listes sont placées dans la liste originale…).

En Pascal

```
Procedure QuickSort(G,D:integer; var A:tableauDeMonType);
var I,J:integer;
    Pivot,X:MonType;
```

```
begin
  I:=G;J:=D;
  Pivot:=A[(G+D) div 2];
  repeat
    while A[i]<Pivot do i:=i+1;
    while A[j]>Pivot do j:=j-1;
    if i<=j then
    begin
      X:=A[i];A[i]:=A[j];A[j]:=X;      {permuter A[i] et A[j]}
      i:=i+1;
      j:=J-1;
    end;
  until i>j;
  if G<j then QuickSort(G,j,A);     {récursion…}
  if i<D then QuickSort(i,D,A);
end;
```

En JavaScript

```
function QuickSort(G,D,A){
i=G;j=D;
Pivot=A[Math.round((G+D)/ 2)];
  do{
    while (A[i]<Pivot) i=i+1;
    while (A[j]>Pivot) j=j-1;
    if (i<=j) {
      X=A[i];A[i]=A[j];A[j]=X;   //permuter A[i] et A[j]
      i=i+1;
      j=j-1;}
    }
while (i<=j);
if (G<j) QuickSort(G,j,A);   //récursion
if (i<D) QuickSort(i,D,A);
};
```

3.9.- Zéro d'une fonction : a - "Recherche binaire"

Pour trouver la valeur d'un "zéro" d'une fonction continue f(x), on peut utiliser l'astuce suivante qui est basée sur la connaissance préalable du comportement de la fonction f(x), supposée continue. Si l'on connaît deux valeurs de x, a et b, telles que le signe de f(a) soit différent de celui de f(b), on sait qu'il existe au moins une valeur de x comprise entre a et b pour laquelle f(x) s'annule. Pour la trouver on utilise le processus de la recherche binaire qui consiste à

1.- évaluer x:=(a+b)/2 (milieu de [a..b]) et calculer f(x),

2.- si f(x) a le même signe que f(a), remplacer a par x, sinon remplacer b par x,

3.- reprendre au point n°1 jusqu'à ce que l'intervalle [a..b] soit suffisamment petit (ce qui fixe le zéro…).

Ceci peut se coder comme suit.

En Pascal

```
function Zero(a,b:real):real;
const eps=1E-8;
var x:real;
begin
  repeat
    x:=(a+b)/2;
    if f(a)*f(x)>0 then a:=x else b:=x;
  until abs(a-b)<eps;
  Zero:=x;
end;
```

En JavaScript

```
function Zero(a,b){
eps=1E-10;
do{
x=(a+b)/2;
if( (f(a)*f(x))>0)  {a=x} else{ b=x};
}while (Math.abs(a-b)>eps);
return(x);
}
```

Notons que ce raisonnement peut facilement être adapté à la recherche d'un élément d'une liste ordonnée de nombres ou de chaînes de caractères. Selon que l'élément recherché est plus grand ou plus petit que l'élément situé au milieu de la liste, on poursuit récursivement la recherche sur une nouvelle liste formée par la moitié supérieure ou inférieure de la liste originale jusqu'à ce que l'on ait trouvé (ou que la nouvelle liste soit réduite à un seul élément, ce qui indique que l'élément recherché n'était pas dans la liste…).

3.10.- Zéro d'une fonction : b - "Méthode de Newton"

La méthode précédente à l'avantage d'être simple et directe mais la convergence vers le zéro est relativement lente. La méthode de Newton-Raphson accélère la convergence en tenant compte de la pente de la fonction pour estimer une nouvelle valeur de l'argument qui se rapproche du zéro de la fonction. A partir d'une valeur initiale (bien choisie) x_1 de l'argument, on construit une suite de x_i par la formule suivante (le ' désigne la dérivée par rapport à x de la fonction f) :

$$x_{i+1} = x_i - f(x_i)/f'(x_i)$$

Cette suite converge (généralement) vers un zéro de f(x). Les problèmes qui peuvent surgir sont de deux ordres : 1) il faut connaître (ou estimer) la dérivée de f, 2) il arrive qu'un mauvais choix de la valeur initiale de x conduise à une série divergente. Avec un peu d'astuce, on peut néanmoins appliquer cette méthode avec succès à la plupart des cas rencontrés en pratique.

Exemple : calcul d'une racine cubique de a : $f(x)=x^3-a$; $f'(x)=3x^2$

En Pascal

```
const eps=1E-9;
 ...
 x:=a/3;
 while abs(x*x*x-a)>eps do x:=x-((x*x*x-a)/(3*x*x));
```

En JavaScript

```
function Rac3(a){//racine cubique d'un nombre
x=a/3;
while (Math.abs(x*x*x-a)>eps) x=x-((x*x*x-a)/(3*x*x));
return(x);
}
y=prompt("Donner un nombre :","");
document.write("La racine cubique de "+y+" est "+Rac3(y));
```

3.11.- Evaluation de fonctions particulières

L'évaluation numérique de certaines fonctions particulières est souvent possible grâce à un développement en série. Celui-ci permet d'obtenir une expression analytique approchée dont les valeurs numériques sont proches de celles de la fonction. Voici quelques exemples tirés du livre d'Abramowitz et Stegun.

Fonction arcsin et arcos

$$\arcsin x = \arctan \frac{x}{\sqrt{1-x^2}} \quad \text{et} \quad \arccos x = \arctan \frac{\sqrt{1-x^2}}{x}$$

Fonction d'erreur

$$P(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-\frac{t^2}{2}} dt = 1 - \frac{1}{2}(1 + C_1 + C_2 x^2 + C_3 x^3 + C_4 x^4)^{-4} \quad \text{pour} \quad x \geq 0$$

avec

$C_1 = 0.196854$

$C_2 = 0.115194$

$C_3 = 0.000344$

$C_4 = 0.019527.$

Si $x<0$, on utilise $P(x) = 1-P(-x)$. (Erreur inférieure à $2.5\ 10^{-4}$)

Fonction d'erreur inverse (soit la valeur de x telle que P(x)=p)

$$x_p = t - \frac{a_0 + a_1 t}{1 + b_1 t + b_2 t^2} \quad \text{avec} \quad t = \sqrt{ln(\frac{1}{1-p^2})} \quad \text{si} \quad 0.5 \leq p \leq 1$$

avec

$a_0 = 2.30753$

$a_1 = 0.27061$

$b_1 = 0.99229$

$b_2 = 0.04481$ (Erreur inférieure à $3\ 10^{-3}$).

Fonction Gamma (extension de la factorielle)
Ramener x dans l'intervalle [0..1] en utilisant $\Gamma(x):=(x-1)\Gamma(x-1)$ puis évaluer

$$\Gamma(x+1) = x! = 1 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 \quad \text{pour} \quad 0 \leq x \leq 1$$

avec

$a_1 = -0.5748646$
$a_2 = +0.9512363$
$a_3 = -0.6998588$
$a_4 = +0.4245549$
$a_5 = -0.1010678$ (Erreur inférieure à $5\ 10^{-5}$).

Dans la référence précitée, qui constitue une référence classique pour ce type de problème, on peut trouver des approximations pour des centaines de fonctions les plus diverses comme, par exemple, les fonctions de Bessel, hypergéométriques, d'Hermite, de Laguerre, trigonométriques, etc…

4.- L'AVENIR…

Il faut savoir qu'il existe des programmes "tout faits" qui résolvent d'une façon automatique les problèmes qui leur sont présentés. L'utilisateur introduit les expressions analytiques des équations qui gouvernent son problème, puis le système, grâce à des techniques sophistiquées d'analyse numérique et syntaxique, comprend le problème et propose des solutions. Ces outils sont arrivés à maturité, leur usage s'est généralisé et révolutionne les méthodes de travail du scientifique!

C H A P I T R E XI

Analyse statistique des données expérimentales

1.- INTRODUCTION

Nous abordons le problème du traitement par ordinateur des données expérimentales. Nous allons adopter une approche essentiellement pragmatique. Nous présentons quelques recettes simples basées sur les statistiques élémentaires qui peuvent aider le scientifique à analyser ses données et à les comparer aux modèles théoriques. Nous supposerons toujours que les conditions "mathématiques" particulières nécessaires à l'application de certaines des formules présentées ci-après sont satisfaites.

2.- MOYENNE ET ECART TYPE

2.1.- Définitions

Lorsque l'on effectue des mesures répétées de la "même" quantité, les valeurs obtenues sont souvent légèrement différentes d'une mesure à l'autre. On peut calculer la **moyenne** de ces résultats en divisant la somme des mesures par leur nombre. La moyenne représente une estimation de la valeur exacte que l'on cherche à mesurer.
La fiabilité de cette estimation dépend de la dispersion des mesures. L'analyse des écarts entre les valeurs mesurées et la moyenne que l'on vient de calculer permet de chiffrer cette dispersion. On appelle résidus les différences entre les valeurs mesurées et leur moyenne. La somme des résidus est nulle mais la somme des carrés des résidus est toujours positive et est d'autant plus petite que les mesures sont peu dispersées autour de la moyenne. La racine carrée du quotient de cette somme de carrés et du nombre de mesures moins un définit une grandeur que l'on appelle l'**écart type** ou l'erreur standard **(σ)** (c'est la valeur que donnent les calculettes douées de fonctions statistiques). C'est une estimation de l'erreur qui entache chacune des mesures que l'on vient de réaliser.
L'écart type sur la moyenne (σ_μ) est égal à cette valeur divisée par la racine carrée du nombre de mesures.

Soit m mesures ayant donné les valeurs a_i. On définit la moyenne par

$$\bar{a} = \frac{1}{m} \sum_{i=1}^{m} a_i$$

et l'écart type (σ) par

$$\sigma = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(a_i - \bar{a})^2}$$

Après quelques calculs, on obtient l'expression équivalente (plus simple à évaluer)

$$\sigma = \sqrt{\frac{1}{m-1}\left(\sum_{i=1}^{m} {a_i}^2 - m\,\bar{a}^2\right)}$$

L'écart type sur la moyenne (σ_μ) est plus petit et vaut

$$\sigma_\mu = \frac{\sigma}{\sqrt{m}}$$

Donc le résultat de nos m mesures peut se synthétiser sous la forme

$$\bar{a} \pm \sigma_\mu$$

2.2.- Mesures d'inégale précision : facteur de poids

Si l'on mesure la même quantité dans des conditions différentes de précision, on peut calculer une moyenne et une erreur sur cette moyenne en affectant à chaque mesure (et à son erreur) un facteur de poids. Si l'on dispose de m mesures $a_i \pm \sigma_i$ on affecte habituellement à chaque mesure un facteur de poids p_i qui est inversément proportionnel au carré de l'écart type $p_i = 1/{\sigma_i}^2$ pour la mesure i. La moyenne et l'écart type sur cette moyenne sont donnés par :

$$\bar{a} = \frac{\sum_{i=1}^{m} \frac{a_i}{\sigma_i^2}}{\sum_{i=1}^{m} \frac{1}{\sigma_i^2}} \quad \text{et} \quad \sigma_\mu = \sqrt{\frac{1}{m-1}\frac{\sum_{i=1}^{m} \frac{(a_i-\bar{a})^2}{\sigma_i^2}}{\sum_{i=1}^{m} \frac{1}{\sigma_i^2}}}$$

Notons que si tous les σ_i sont égaux, on retrouve bien la formule qui donne l'écart type sur la moyenne.

2.3.- Erreur sur des comptages

Nous considérons ici les mesures qui résultent d'un comptage d'événements fortuits (comme les photons qui tombent sur un détecteur de lumière ou la détection de particules radioactives). L'écart type sur le nombre N_c d'évènements comptés (lorsque celui-ci est suffisamment grand : en pratique >≈30) vaut la racine carrée du nombre (cf. statistique de Poisson).

Par exemple, si, pour un intervalle de temps déterminé, on détecte 1000 photons, l'écart type sur cette mesure sera ≈ ±32.

Erreur sur comptage :

$$\sigma = \sqrt{N_c}$$

et

$$N_c \pm \sqrt{N_c}.$$

2.4.- Signification de l'erreur

L' “erreur” calculée comme nous venons de l'expliquer apporte une information importante sur la qualité des mesures mais ne permet évidemment pas de garantir à 100% que la valeur que l'on cherche à mesurer est comprise entre $\bar{a} + \sigma_\mu$ et $\bar{a} - \sigma_\mu$. La théorie des statistiques nous permet seulement d'affirmer que la valeur recherchée est comprise entre $\bar{a} + \sigma_\mu$ et $\bar{a} - \sigma_\mu$ avec une probabilité de 68% seulement. Il y a donc ≈1/3 de chance que cette valeur soit hors de la fourchette proposée! Pour augmenter cette probabilité, il faut agrandir la fourchette.
Si l'on double l'erreur ($\pm 2\sigma_\mu$), la probabilité passe à 95,45% et il n'y a plus qu'une chance sur 22 que la valeur exacte soit hors de cette fourchette (et il n'y a qu'une une chance sur ≈1,7 millions que la valeur recherchée ne soit pas dans la fourchette correspondant à $\pm 5\sigma_\mu$).
Donc le calcul de la moyenne et de l'erreur permet de synthétiser un grand nombre de mesures sous une forme simple.

3.- LES TESTS D'HYPOTHESE

3.1.- Utilité et signification

Les tests d'hypothèses sont conçus pour répondre aux questions que se posent les chercheurs lorsqu'ils analysent les données issues d'expériences ou de relevés. Les statisticiens ont développé un arsenal impressionnant de tests qui s'adaptent chacun à des situations bien particulières. Nous allons ci-après en évoquer deux des plus utiles (et des plus simples). Le test dit du “t de Student” et le test du “chi-carré”. Le premier permet de mesurer l'impact d'un paramètre sur une expérience et l'autre d'évaluer l'accord entre des prédictions d'un modèle et des mesures expérimentales.
La conclusion d'un test d'hypothèse est habituellement une probabilté : la probabilité que l'hypothèse avancée soit (ou ne soit pas) vérifiée. On considère généralement qu'une hypothèse est significative si la probabilité qu'elle soit vraie est supérieure à 95%. On considère également que l’hypothèse est parfaitement confirmée si la probabilité d'un accord "par hasard" est de moins de 1%.
Notons ici que l'on utilise souvent le fait que la probabilité de l'hypothèse opposée vaut 1 moins celle de l'hypothèse directe. Par exemple, si la probabilité que deux valeurs puissent être égales "par hasard" est très faible cela implique que les deux valeurs sont significativement différentes.

3.2.- t de Student

La stratégie habituelle pour étudier l'influence d'un paramètre particulier sur une expérience est de réaliser deux séries de mesures pour des valeurs différentes de ce paramètre. Pour chacune de ces deux séries, on peut calculer une moyenne (et l'erreur sur cette moyenne). Si les moyennes diffèrent fortement d'une série à l'autre, on peut immédiatement conclure que le paramètre influence l'expérience. Cependant, si les moyennes sont seulement légèrement différentes, il devient délicat de conclure. C'est alors qu'il faut faire appel au test statistique dit du “t de Student” qui permet de vérifier si deux séries de mesures sont “significativement” différentes.
L'application du test impose certaines contraintes qui sont généralement satisfaites si les mesures ont été réalisées avec le même appareillage et si l'effet du paramètre que l'on fait varier est faible.

3.2.1.- Calcul numérique du t de Student

a) Deux groupes distincts de mesures

Pour appliquer le test, il faut calculer t qui est défini comme la différence des moyennes divisée par l'écart type sur cette différence (qui est calculé comme la racine carrée de la somme des carrés des écarts types).
Si l'on dispose de deux séries de mesures évaluées indépendamment l'une de l'autre, les a_i (m_a mesures) et les b_i (m_b mesures), on peut calculer t à partir des moyennes et des erreurs de chaque série par la relation :

$$t = \frac{\bar{a} - \bar{b}}{\sqrt{\sigma_{\mu_a}^2 + \sigma_{\mu_b}^2}}$$

A partir d'une formule mathématique compliquée, on peut alors calculer la probabilité que la différence observée entre les deux moyennes soit significative (et non pas due simplement à la dispersion des mesures).
Cette formule fait intervenir le “nombre de degrés de liberté” ν du système qui est égal au nombre total de mesures moins deux.

$$\nu = m_a + m_b - 2$$

b) Mesures pairées

Si les mesures ont été réalisées par paires, c'est-à-dire que, pour un même échantillon, on a réalisé m paires (a_i;b_i) de mesures en changeant chaque fois les conditions expérimentales, on peut calculer t à partir des différences $d_i = a_i - b_i$.
Soit la moyenne de ces différences et l'écart type sur cette moyenne. On a alors

$$t = \frac{\bar{d}}{\sigma_{\mu_d}}$$

avec

$$\nu = m - 1$$

Attention, le degré de liberté à considérer est m-1 (non pas m-2!).

c) Echantillon unique

Le test de Student peut aussi être utilisé pour comparer une unique série de mesures réalisées sur un échantillon dont on connaît parfaitement bien la moyenne. Dans ce cas, il suffit de remplacer la moyenne du second groupe de mesures par cette valeur et de considérer que sa dispersion est nulle (σ=0). Le degré de liberté à considérer est encore le nombre de mesures moins 2.

3.2.2.- Estimation numérique du t de Student

Si l'on connaît t et ν, le programme suivant (en Pascal) calcule la probabilité (entre 0 et 1) de ne pas obtenir "par hasard" la valeur calculée de t. Donc si cette probabilité est grande, cela signifie que la valeur de t qu'on a déduite des observations ne peut pas être simplement due au hasard. On a bien une différence significative entre les groupes de mesures et, par exemple, si la probabilité dépasse 0.95, on peut conclure qu'on observe une différence très significative.
Le calcul repose sur l'évaluation numérique d'un développement en série. (Réf. Handbook of mathematical Functions, Abramowitz & Stegun, National Bureau of Standards, 1968).

En Pascal

```
function Student(t: real; nu:integer): real;
  var
   a,s,x:real;
 begin
  t:=arctan(abs(t)/sqrt(nu));
  s:=0; a:=nu-3;
  x:=sqr(cos(t));
  while a > 0 do
   begin
    s:=(s + 1) * (a/(a + 1)) * x;
    a:=a-2;
   end;
  if nu <> 1 then s:=(s+1) * sin(t);
  if odd(nu) then s :=(s*cos(t)+ t)* 2/pi;
  Student := s;
 end;
```

En JavaScript

```
function Student(t,nu){
  t=Math.atan(Math.abs(t)/Math.sqrt(nu));
  nu=Math.round(nu);
  s=0; a=nu-3; x=Math.cos(t);x=x*x;
  while (a > 0){s=(s + 1)*(a/(a+1))*x;a=a-2;}
  if (nu != 1) {s=(s+1) * Math.sin(t)};
  if ((nu % 2) !=0) {s=(s*Math.cos(t)+ t)* 2/Math.PI};
  return(s)
};
```

3.2.3.- Exemples numériques

a) Deux groupes de valeurs non pairées
(Ex : Influence de la température sur la croissance d'une plante)
On mesure deux groupes de valeurs :

	Situation A	Situation B
	12	19
	15	17
	8	12
	14	17
	12	20
	10	13
	16	18
	18	
	10	
Moyenne	12.43	15.78
±sigma	1.07	1.13

t = 2.158
nu = 14
Probabilité d'une différence = 95.12% (significatif)

b) Un groupe de valeurs pairées
(Ex : Variation de poids des mêmes animaux soumis à des régimes différents)
On mesure les valeurs suivantes :

Echantillon	Situation A	Situation B	Différence
N°1	25	29	4
N°2	12	13	1
N°3	17	18	1
N°4	21	20	-1
N°5	9	15	6
Moyenne des différences		2.2	
±Sigma		1.24	

t = 1.77
nu = 4
Probabilité d'une différence = 84.8% (pas vraiment significatif)

c) Un groupe de mesures comparées à une moyenne bien connue
(Ex : légère modification d'un appareil qui produit en moyenne un certain nombre de pièces.)

La moyenne "bien connue" est de 45. On modifie légèrement un paramètre et on mesure les valeurs suivantes :

46
47
48
47
47

Moyenne	47
±Sigma	0.32

t = 6.32
nu = 3
Probabilité d'une différence = 99.2% (vraiment significatif)

4.- AJUSTEMENT D'UNE COURBE PARAMETRIQUE

4.1.- La méthode dite "des moindres carrés"

Un problème classique en sciences consiste à comparer des valeurs expérimentales avec celles prédites par un modèle mathématique adéquat. Le modèle conduit généralement à une expression analytique qui contient certains paramètres qui devront être évalués en fonction des résultats des expériences. La technique dite "des moindres carrés" est habituellement mise en oeuvre pour calculer les valeurs numériques des paramètres qui assurent le "meilleur accord" entre le modèle et l'expérience.

Soit f ($x,c_1..c_n$) la fonction qui dépend de la variable x et de n paramètres $c_1..c_n$. Soit y_i les résultats de m (m > n) mesures effectuées pour certaines valeurs connues x_i de la variable. On définit l'expression

$$S = \sum_{i=1}^{m} [f(x_i, c_1, ..., c_n) - y_i]^2$$

qui servira à mesurer l'accord entre le modèle et l'expérience. Cette somme de carrés est toujours positive et sera d'autant plus petite que les valeurs de la fonction f en x_i seront proches des y_i. Pour trouver les "meilleures valeurs" des paramètres, on

recherche le minimum de l'hypersurface définie par l'expression de S dans l'espace à n+1 dimensions formé par les $c_1..c_n$ et x. D'une façon imagée, cette recherche consiste à descendre les "vallées" de l'hypersurface pour découvrir un “trou” qui correspond à un minimum de S et dont les coordonnées donnent un jeu de $c_1..c_n$ qui assurent le meilleur accord entre le modèle et l'expérience. Le codage des algorithmes de recherche du minimum de l'hypersurface fait appel à des astuces qui sont implémentées dans différents programmes (comme WinCurvefit par ex.).
Notons aussi que l'ajustement par moindres carrés permet sous certaines conditions d'estimer la probabilité que l'expression analytique proposée décrive bien le phénomène observé (méthode du chi-carré, cf. §4.3), ce qui peut quelquefois aider à discriminer différents modèles décrivant la même expérience.

4.2.- Ajustement d'une fonction polynomiale : Régression

Lorsque la fonction à ajuster est un polynôme, la méthode des moindres carrés mène à un système d’équations relativement simple à résoudre analytiquement. Il est alors possible de calculer directement, à partir des données, les valeurs optimales des coefficients du polynôme qui garantit le meilleur accord avec les données.
Nous allons donner les solutions dans le cas d’une droite et d’une parabole ((polynôme d’ordre 1 et 2).

Les expressions suivantes permettent de calculer les deux coefficients a et b de la “meilleure” droite du type y=ax+b passant par un ensemble de points (x_i,y_i), i=1,2,…,n

$$a = \frac{\sum_{i=1}^{n} x_i y_i - \frac{\left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n}}{\sum_{i=1}^{n} x_i^2 - \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n}}$$

$$b = \frac{\sum_{i=1}^{n} y_i}{n} - a\frac{\sum_{i=1}^{n} x_i}{n}$$

Le coefficient de détermination est donné par

$$r^2 = \frac{\left[\sum_{i=1}^{n} x_i y_i - \frac{\left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n}\right]^2}{\left[\sum_{i=1}^{n} x_i^2 - \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n}\right]\left[\sum_{i=1}^{n} y_i^2 - \frac{\left(\sum_{i=1}^{n} y_i\right)^2}{n}\right]}.$$

Il est toujours inférieur ou égal à 1 et mesure le degré de perfection de l'ajustement de la droite de régression. Pour que l'ajustement puisse être considéré satisfaisant, il faut généralement que r soit supérieur à 0,9.

Les expressions suivantes permettent de calculer les trois coefficients a, b, c, de la “meilleure” parabole du type $y = ax^2 + bx + c$ passant par un ensemble de plus de 2 points : (x_i,y_i), i=1,2,…,n :

$$a = \frac{A - B}{\left[n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2\right]\left[n\sum_{i=1}^{n} x_i^4 - \left(\sum_{i=1}^{n} x_i^2\right)^2\right] - \left[n\sum_{i=1}^{n} x_i^3 - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} x_i\right)^2\right]^2}$$

avec

$$A = \left[n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2\right]\left[n\sum_{i=1}^{n} x_i^2 y_i - \left(\sum_{i=1}^{n} x_i^2\right)\left(\sum_{i=1}^{n} y_i\right)\right]$$

$$B = \left[n\sum_{i=1}^{n} x_i^3 - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} x_i^2\right)\right]\left[n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)\right]$$

$$b = \frac{\left[n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)\right] - a\left[n\sum_{i=1}^{n} x_i^3 - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} x_i^2\right)\right]}{\left[n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2\right]}$$

$$c = \frac{1}{n}\left[\sum_{i=1}^{n} y_i - a\sum_{i=1}^{n} x_i^2 - b\sum_{i=1}^{n} x_i\right]$$

Pour les ajustements de polynômes de degré plus élevé, il est encore possible de calculer directement les meilleurs paramètres mais les expressions deviennent de plus en plus compliquées à calculer et ne seront pas données ici.

4.3.- Validité des ajustements d'observations : Test du chi-carré

Le test du chi-carré est habituellement lié à l'étude des corrélations qui peuvent exister entre des groupes ou des classes de choses. Nous allons cependant négliger cet aspect statistique pour nous limiter à l'étude d'un problème particulier qui est celui de l'ajustement d'un modèle paramétrique à des données expérimentales. Nous allons plus particulièrement discuter la relation qui existe entre la méthode dite des moindres carrés et la distribution du chi-carré.

Le problème est le suivant. On dispose d'un modèle théorique qui décrit le phénomène que l'on va observer sous forme d'une expression analytique qui dépend de quelques paramètres. Pour tester le modèle, on réalise une série de mesures pour lesquelles on fait varier les valeurs de un (ou plusieurs) de ces paramètres d'une façon précise. Les valeurs des autres paramètres sont inconnues mais on s'arrange pour qu'ils ne varient pas entre les différentes mesures. Si le nombre de mesures est suffisant, on peut :
a- calculer les “meilleures” valeurs qu'il faut donner aux paramètres indéterminés pour correspondre à nos mesures,
b- calculer la probabilité que les fluctuations entre nos mesures et les prédictions du modèle soient uniquement dues aux erreurs statistiques. Si cette probabilité est grande cela signifie que le modèle décrit adéquatement notre expérience.
La stratégie à suivre pour répondre au premier point est celle dite des moindres carrés.
Supposons que pour n valeurs x_i du paramètre x nous ayons mesuré les valeurs m_i affectées chacune d'une erreur σ_i. Le modèle propose une expression analytique $f(x,c_1,...,c_m)$ qui dépend du paramètre x (qui varie) et de m autres paramètres (les c_i qui, dans notre expérience, sont gardés constants).

L'expression suivante, qui définit le chi-carré,

$$\chi^2 = \sum_{i=1}^{n} \left(\frac{f(x_i, c_1, \ldots, c_m) - m_i}{\sigma_i} \right)^2$$

est une somme de carrés (toujours positive) dont la valeur est d'autant plus petite que les différences entre les valeurs mesurées m_i et les valeurs calculées f(x_i…) sont faibles. Sans entrer dans des considérations théoriques, nous admettrons que les valeurs des c_i qui minimisent cette expression sont les meilleures. Pour les obtenir, on dispose de tout un arsenal de processus mathématiques qui permettent de rechercher le minimum d'une hypersurface dans un espace à $m+1$ dimensions (les m c_i plus la valeur de la fonction). Ce minimum définit un jeu de c_i qui seront nos va-

leurs optimales. La valeur résiduelle de l'expression (qui n'est généralement pas nulle) va servir à évaluer la qualité du modèle. C'est ici qu'intervient le test d'hypothèse du chi-carré.

On peut montrer que la fonction considérée est distribuée comme un chi-carré à v= $n–m–1$ degrés de liberté. La valeur la plus probable de cette fonction est celle qui correspond au maximum de la courbe de distribution, soit une probabilité de 50%. C'est ce qui doit se produire si les erreurs entre les valeurs calculées et mesurées sont dues principalement à des phénomènes statistiques. Le programme présenté ci-après permet de calculer précisément cette probabilité.

4.3.1.- Estimation de la valeur numérique du chi-carré

Un raisonnement intuitif suggère que le chi-carré doit avoir une valeur proche de n-m car, en moyenne, l'erreur entre f(x_i…) et m_i est ≈ σ_i donc chacun des termes de la somme est ≈1 et l'expression considérée devrait valoir ≈n. Si l'on admet, de plus, que chacun des paramètres permettrait d'"annuler" au moins un des termes de la somme, cela conduit à une valeur finale de n-m proche de la valeur théorique v= $n–m–1$. Donc, une façon simple de vérifier la qualité d'un ajustement, est de vérifier que la valeur du chi-carré divisé par v(=$n–m–1$) est proche de 1 (on appelle cette valeur le chi-carré réduit). On considère habituellement que l'ajustement est convenable lorsque la valeur du chi-carré réduit est comprise entre 0.5 et 1,5.

4.3.2.- Comment savoir si le nombre de paramètres est optimum?

Certains modèles permettent d'ajouter (ou de soustraire) un ou plusieurs paramètres dans l'expression analytique (par exemple on peut augmenter le degré d'un ajustement polynomial). Pour évaluer le bien fondé de l'ajout d'un paramètre on peut effectuer les calculs suivants.

a- Calculer le chi-carré à m-1 paramètres.

b- Calculer le chi-carré à m paramètres (qui doit être plus petit).

c- Evaluer l'expression

$$\sqrt{\frac{\chi^2(m-1)-\chi^2(m)}{\frac{\chi^2(m)}{n-m-1}}}$$

Cette expression est distribuée comme un t de Student à n-m-1 degrés de liberté, ce qui permet de calculer la probabilité que l'adjonction du paramètre soit justifiée (cf. Bevington p.202 et A&S p.947).

4.3.3.- Estimation numérique du chi-carré

Le programme suivant (en Pascal) calcule pour un degré de liberté (nu) et un chi-carré (chi) donnés, la probabilité (entre 0 et 1) d'observer cette valeur du chi-carré. La valeur la plus probable du chi-carré est, en théorie, celle qui correspond à une

probabilité de 50%. Une valeur de la probabilité qui s'écarte fortement de cette valeur moyenne indique généralement que le désaccord entre le modèle et les mesures n'est pas dû uniquement au hasard. Notons ici que si la probabilité est très petite (ce qui correspond à un chi-carré très petit) cela indique que, soit les erreurs sur les mesures sont largement surestimées, soit que le modèle ajuste trop bien les données pour être honnête.
Le calcul repose sur l'évaluation numérique d'un développement en série. (Réf. Handbook of mathematical Functions, Abramowitz & Stegun, National Bureau of Standards, 1968).

En Pascal

```
function ChiSqr(chi:real;nu:integer):real;
 var
   a,s,g:real;
   k:integer;
 begin
 chi:=abs(t); s:=0.0; a:=1.0; g:=1.0; k:=0;
 while g <> s do
   begin
    g:=s; s:=s+a;
    k:=k+2;
    a:=a * chi/(nu + k);
   end;
  a:=nu/2.0;
  while a > 0 do
   begin
    s:=s/a;
    a:=a-1.0;
   end;
  if odd(nu) then s:=s/sqrt(pi);
  chi:=chi/2.0;
  s:=s * exp(-chi+(nu/2)*ln(chi));
  ChiSqr:=s;
 end;
```

En JavaScript

```
function ChiSqr(chi,nu){
 nu=Math.round(nu);
 chi=Math.abs(chi);s=0.0;a=1.0;g=1.0;k=0;
 while (g != s){
    g=s; s=s+a;k=k+2;
    a=a*chi/(nu+k);
    }
  a=nu/2.0;
  while (a > 0) {
    s=s/a;a=a-1.0;
  }
  if ((nu%2)!=0){s=s/Math.sqrt(Math.PI);}
  chi=chi/2.0;
  s=s * Math.exp(-chi+(nu/2)*Math.log(chi));
  return(s);
}
```

4.3.4.- Exemples numériques

Il s'agit de l'intensité lumineuse mesurée en fonction du temps qui s'est écoulé après une excitation. La théorie prévoit une superposition de décroissances exponentielles.

Les données

temps (sec)	Intensité (nb. photons)
0	5695
1	2348
2	1126
3	675
4	424
5	328
7	244
9	173
12	94
15	65

Ajustement par une somme d'exponentielles décroissantes du type

$$\sum_{i=1}^{n} A_i \mathrm{e}^{-t/k_i}$$

Nb. param.	chi-carré	nu	chi réduit	fchi	t	Prob	
1	7920	8	990	100%			
2	587	7	84	100%	87	100%	justifié
3	168	6	28	100%	15	99,99%	justifié
4	3.45	5	0.69	37%	232	100%	justifié
5	3.24	4	0.81	48%	0.26	19%	inutile
6	3.05	3	1.02	62%	0.19	14%	inutile

Résultat de l'ajustement avec 4 paramètres (cf. ExpFit) :

A1	5001.6
k1	0.96
A2	689.5
k2	6.29

C H A P I T R E XII

Fonctionnement des ordinateurs

Aspect technique

1.- COMPOSITION D'UN ORDINATEUR

La figure 1 ci-dessous représente le schéma très simplifié d'un ordinateur. En plus du processeur, tous les ordinateurs comportent au minimum une horloge, une mémoire, quelques périphériques et deux bus (données et adresses) qui permettent les liaisons électriques entre tous ces éléments. Il faut aussi une alimentation stabilisée qui fournit la puissance à tous les composants et un coffret.

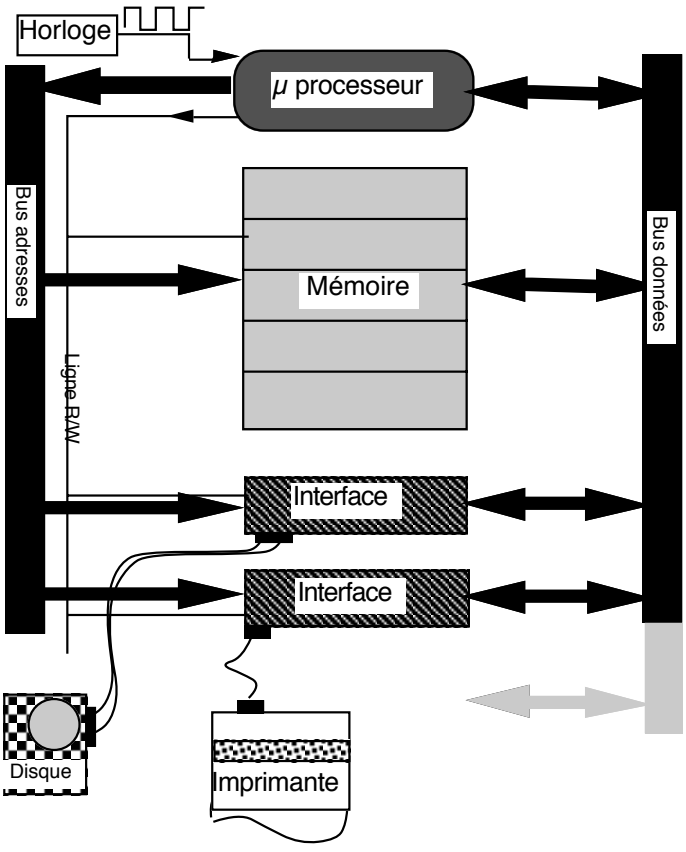
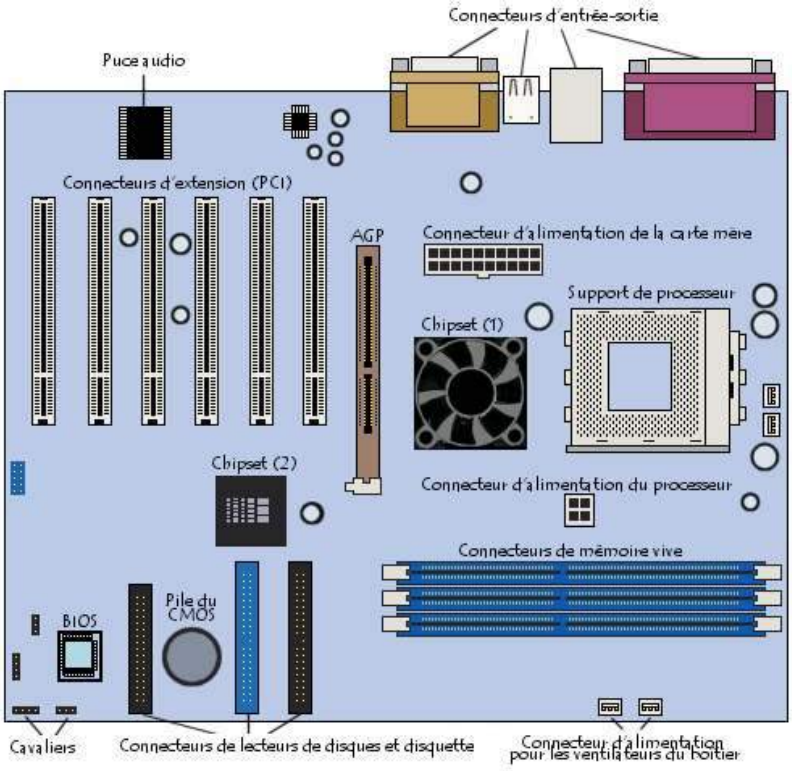


Fig. 1. Schéma de principe d'un ordinateur

Le bus des adresses permet au processeur d'indiquer le numéro de la cellule mémoire avec laquelle il désire échanger une information. Le bus des données véhicule cette information dans le sens indiqué par la ligne lecture/écriture (ou R/W Read/

Write). Le sens lecture va de la mémoire vers le processeur, le sens écriture du processeur vers la mémoire.
Les liaisons entre les différents éléments et les bus sont réalisées par des circuits à trois états (tri-states). Ces circuits peuvent être positionnés, soit en mode écriture (ils imposent les niveaux électriques des fils du bus), soit en mode lecture (ils lisent l'état du bus), soit en mode passif (ils s'isolent du bus). En pratique, un seul circuit à la fois est en mode écriture, les autres sont soit passifs, soit lecteurs. (Si, par erreur, plusieurs circuits tentent d'écrire en même temps, il y a généralement destruction physique du plus faible…). Le mode passif est l'état normal des circuits de liaison, ceux-ci changent de mode uniquement pendant le temps nécessaire à un échange d'informations.
Notons que les interfaces des périphériques se comportent généralement comme n'importe quelle zone de mémoire et que, pour converser avec un périphérique, le processeur lit et écrit dans l'interface exactement comme il le ferait dans la mémoire.
Il existe sur le marché un très grand nombre de processeurs ayant tous des caractéristiques spécifiques (cf. http://www.cpu-world.com/CPUs/CPU.html). Une explication du fonctionnement d'un ordinateur et de ses périphériques est donnée en http://www.commentcamarche.net/pc/pc.php3 .

Les différents éléments qui composent un ordinateur sont placés sur circuit imprimé de grande taille. Ce circuit constitue la "carte mère" et permet la connexion des éléments essentiels qui forment l'ordinateur (cf. Fig. 2).



Repris de : http://www.commentcamarche.net/pc/images/motherboard.png
Fig. 2. Disposition physique des éléments sur une carte mère

1.1.- Les registres

Le processeur contient un certain nombre de registres qui peuvent chacun mémoriser une valeur binaire. Le nombre de bits qui composent un registre peut varier de 8 à 80. Le processeur 8088 de l'IBM-PC (1979) contenait 14 registres de 16 bits, un PENTIUM (2005) possède plus de 200 registres de 32 bits.
Les registres constituent des “emplacements mémoire” privilégiés, situés directement dans le processeur. Ce dernier peut donc y accéder très rapidement et y effectuer des opérations logiques ou arithmétiques complexes qui ne sont pas réalisables directement dans la mémoire de l'ordinateur. Les données sont généralement lues depuis la mémoire vers un registre, modifiées puis réécrites dans la mémoire. Par exemple, l'instruction A=B+C; sera traduite par la séquence suivante :

- Le contenu de B est placé dans un registre libre.
- Le contenu de C est additionné à ce registre.
- Le contenu du registre est écrit en mémoire à l'adresse occupée par A.

En langage assembleur, cette séquence d'instructions pourrait s'écrire :

```
MOV #R3,B
ADD #R3,C
MOV A,#R3
```

1.2.- Le registre compteur

Parmi les registres du processeur, il en est un qui joue un rôle fondamental, c'est le registre compteur (Program Counter ou PC). Il contient l'adresse de l'instruction que va exécuter le processeur. Ce registre est automatiquement incrémenté après chaque instruction.
Un déroutement du programme (GO TO ...) revient habituellement à placer, dans ce registre, l'adresse de la première instruction de la nouvelle séquence que l'on veut exécuter.

1.3.- Principe du séquencement des instructions

L'horloge interne génère des impulsions électriques qui contrôlent tout le fonctionnement de l'ordinateur. La fréquence du train d'impulsions est directement liée à la vitesse d'exécution du processeur. Les fréquences utilisées sur la plupart des micro-ordinateurs allaient de 1MHz (Commodore 64 en 1977) à 3 à 4 GHz (Pentium en 2005). A chaque battement de l'horloge, correspond l'exécution d'une instruction élémentaire. Cette exécution se décompose en deux phases.
- Pendant la première phase, tous les circuits du processeur (et certains circuits externes) se positionnent en fonction de l'instruction qui va devoir être exécutée. (Les niveaux électriques changent.)
- Pendant la seconde phase, les circuits de l'ordinateur mémorisent l'état des différents signaux. (Tous les niveaux électriques sont stables.)

Le déroulement d'un programme se décompose comme suit.

1. Lecture d'une instruction

a) Phase de positionnement

Le processeur place le contenu du registre compteur sur le bus des adresses et indique (ligne R/W) qu'il désire effectuer une lecture. Ceci amène les circuits de la mémoire à placer le contenu de la case demandée sur les fils du bus des données.

Le processeur connecte le registre d'instruction avec le bus des données.

b) Phase de mémorisation

Le processeur fige l'état du registre d'instruction et incrémente le registre compteur d'une unité.

2. Exécution de l'instruction

Le processeur positionne ses circuits en fonction de l'action demandée.

S'il s'agit d'une opération interne (modification du contenu d'un registre, déroutement, etc…), les bus et la ligne R/W ne participent pas à l'opération. Si l'instruction concerne la mémoire (lecture/écriture etc…), le processeur adapte l'état des bus à l'opération demandée.

Exemple 1 : Mise à zéro d'un registre

a) Phase de positionnement

Le processeur positionne ses circuits internes de façon à annuler tous les bits du registre concerné.

b) Phase de mémorisation

Le contenu du registre est mémorisé.

Exemple 2 : Ecriture du contenu d'un registre dans une case mémoire

a) Phase de positionnement

Le processeur place l'adresse de la case mémoire sur le bus des adresses et le contenu du registre sur le bus des données, tout en indiquant sur la ligne R/W qu'il veut effectuer une écriture. Les circuits de la mémoire connectent la case concernée avec le bus des données.

b) Phase de mémorisation

La mémoire fige l'état de la case concernée qui va donc "mémoriser" l'état du bus des données (qui correspond au contenu du registre).

2.- LES INTERFACES ET LES PROTOCOLES D'INTERFACAGE

Une interface est un dispositif électronique qui relie un ordinateur à d'autres instruments. L'interface adapte les signaux produits par l'ordinateur à ceux de l'autre instrument. Les interfaces sont généralement placées dans la boîte de l'ordinateur (dans des "slots") et comportent un ou plusieurs connecteurs électriques qui seront reliés par des câbles aux divers instruments à commander.

Certaines interfaces utilisent des règles bien définies pour converser avec un instrument. L'ensemble de ces règles forme ce que l'on appelle un protocole d'interfa-

çage. Il porte non seulement sur les formes et les amplitudes des signaux électriques mais aussi sur la logique des échanges entre l'instrument et l'ordinateur.

La construction des interfaces suppose des connaissances précises du fonctionnement intime de l'ordinateur et de l'instrument à connecter. Il faut disposer des manuels techniques adéquats et avoir de très bonnes connaissances en électronique. Nous n'insisterons donc pas sur ce point. Néanmoins, la **programmation des interfaces** est relativement simple et peut souvent se faire à partir d'un langage de haut niveau. Nous allons décrire ci-après les principales techniques utilisées.

Les protocoles d'interfaçage font l'objet de normes édictées par certains organismes officiels (IEEE : Institute of Electrical and Electronical Engineers, EIA : Electronic Industries Association, ANSI : American National Standards Institute, ISO : International Organization for Standardization, etc…). C'est généralement la référence du document les décrivant qui leur donne un nom (cette référence n'a pas de signification particulière : des lettres et des chiffres…).

2.1.- Quelques protocoles d'interfaçage

2.1.1.- EIA-RS232C (liaison série asynchrone)

Le protocole RS232 définit les normes de communication utilisées pour transmettre à distance des informations digitales sur une ligne électrique comportant deux fils et une ligne de terre. La sortie d'un des instruments est reliée à l'entrée de l'autre et réciproquement. Les informations sont codées sous forme de pulses électriques qui sont émis sur la ligne à une fréquence maximale exprimée en **bauds** (nb. max. de pulses par sec). Chaque fois qu'on désire envoyer un octet (8 bits), l'interface transforme ces 8 bits en un train d'impulsions formé comme suit :

- start bit (1),
- 7 ou 8 bits de data (soit 0 ou 1),
- parity bit (facultatif),
- un ou deux stop bit(s) (1).

Le système de détection attend le premier bit de start, puis échantillonne la ligne à intervalles réguliers (en fonction du nombre de bauds de la ligne) et reconstruit l'octet envoyé. Il vérifie aussi que le bit de stop (et celui de parité) sont bien là.
La liaison RS232 est habituellement utilisée pour relier un terminal avec un ordinateur (soit directement, soit par modem : cf §5). Certaines imprimantes (dites "séries") utilisent aussi ce protocole.
AVANTAGES
- Faible coût de la ligne de transmission (deux fils et une masse) et simplicité de connexion.
- Protocole simple, implémenté sur la quasi-totalité des ordinateurs.
INCONVENIENT
- Faible vitesse de transmission.
Les fréquences utilisées sont normalisées, les valeurs les plus courantes sont :
110 bauds : utilisée autrefois sur les télétypes mécaniques (≈10 octets/s),

300 bauds : vitesse des transmissions par modem acoustique sur une ligne téléphonique (≈30 octets/s),
1200 bauds : vitesse des modems reliés directement au réseau téléphonique (≈120 octets/s),
2400 bauds : liaison directe d'un terminal situé à moyenne distance,
4800 bauds : même chose,
9600 bauds : liaison d'un périphérique situé à courte distance (par exemple une imprimante),
14400 bauds & 28800 bauds : fréquences utilisées par les modems récents à hautes performances. L'accès à l'Université de Liège est possible à ces vitesses (cf. n° tél. 04 366 2884).

Note : Protocole de synchronisation dit "X-on/X-off"

Une méthode souvent utilisée pour synchroniser les transmissions séries consiste à réserver deux codes spécifiques pour signaler soit qu'on est prêt à recevoir de nouveaux caractères soit qu'on n'est pas prêt (saturation des tampons par exemple). Lorsqu'un des deux transmetteurs reçoit le code X-off, il doit s'arrêter immédiatement de transmettre et attendre la réception du code X-on pour reprendre l'émission.

Les codes habituellement utilisés sont :

- prêt à recevoir (X-on) : Ctrl-Q (ASCII 17),
- arrêter les envois (X-off) : Ctrl-S (ASCII 19).

Ce mode de synchronisation s'appelle aussi Ctrl-Q/Ctrl-S.

Le protocole RS232 comporte plusieurs autres recommandations et décrit l'utilisation d'autres signaux que ceux évoqués ici. Nous renvoyons le lecteur intéressé aux documents techniques adéquats.

2.1.2.- Centronics parallel

Il s'agit d'un standard introduit par la firme d'imprimantes Centronics Data Computer Corp. Bien que non officiel, il est admis par la grande majorité des constructeurs d'ordinateurs. Il est généralement utilisé pour relier une imprimante (dite "parallèle") à un ordinateur.

La liaison comporte 36 fils. Parmi ceux-ci, 8 sont des fils de données qui permettent de transmettre en parallèle les 8 bits d'un octet. Les autres fils permettent la synchronisation des échanges selon une méthode dite du "hand shaking" (poignée de mains).

1.- L'ordinateur attend que l'imprimante signale qu'elle est prête à recevoir des données (cf. ligne BUSY).

2.- L'ordinateur modifie les états des huit fils de données en y plaçant une image de l'octet à transmettre et envoie un pulse (STROBE) pendant lequel l'imprimante lit l'état des données et reconstruit l'octet (pendant le strobe, l'état des fils de données est stable).

3.- Dès que l'imprimante a fini de traiter les données, elle le signale à l'ordinateur par un pulse (ACKnowledge).

AVANTAGES
- On peut atteindre une très grande vitesse de transmission.
- Interface peu coûteuse à réaliser, protocole simple.
INCONVENIENTS
- Le câble et les connecteurs (qui comportent beaucoup de fils) sont fragiles et difficiles à isoler électriquement.
- Dès que le câble de liaison dépasse 2 à 3 mètres de long, on risque des erreurs de transmission dues aux parasites électriques.

2.1.3.- Bus IEEE488 (GPIB, HPIB,…)

GPIB = General Purpose Interface Bus

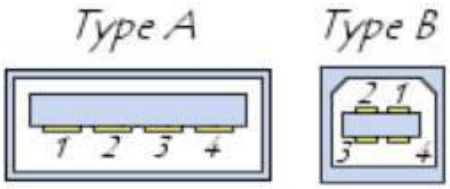
HPIB = Hewlett-Packard Interface Bus

Le standard IEEE488 décrit un mode de liaison entre divers appareils électroniques. Il repose sur un dispositif originalement proposé par la firme Hewlett-Packard.
Il se compose physiquement de câbles à 24 fils ayant des connecteurs spéciaux mâles/femelles enfichables les uns aux autres (daisy chain). Les différents appareils, connectés entre eux par ces câbles, peuvent dialoguer selon un protocole complexe. L'un des appareils orchestre les liaisons, il s'agit du contrôleur (généralement l'ordinateur). Les autres sont tour à tour, soit des écouteurs (listeners), soit le parleur (talker). Chaque appareil a une adresse (un numéro) et ne décode que les messages qui lui sont destinés. Le contrôleur peut s'adresser à l'un ou plusieurs des appareils pour modifier leurs états ou leur poser des questions. Si un appareil veut répondre, il doit attendre l'autorisation du contrôleur qui lui passe la main et attend sa réponse.

AVANTAGES
- Bien adapté aux instruments de laboratoire semi-intelligents (spécialement ceux produits par Hewlett-Packard…).
- On peut très facilement ajouter et retirer des appareils.
- On peut effectuer beaucoup d'opérations sur plusieurs appareils en même temps.
INCONVENIENTS
- Le protocole est complexe et certains constructeurs ne respectent pas toutes les contraintes, ce qui provoque des blocages du bus.
- La distance entre les appareils est limitée (quelques mètres) et le bus est sensible aux parasites électriques.
- Les suppléments à payer pour équiper un appareil aux normes GPIB sont généralement élevés.

2.1.4.- USB (Universal Serial Bus)

Le bus USB (Universal Serial Bus; en français : Bus Série Universel) permet des liaisons “séries” basées sur un seul fil (le bus) plus une masse. En pratique, les connecteurs sont standardisés et comportent 4 fils: une masse, une alimentation (+5V) et deux fils pour les données qui sont transmises en mode différentiel. La présence du +5V permet d’auto-alimenter de petits périphériques (comme une souris). L'architecture en bus permet d’atteindre des vitesses de transmission élevées (de 12 Mbits/s pour l’USB 1) à 480 Mbits/s pour l’USB2).



1. Alimentation +5V (*VBUS*) 100mA maximum ; 2. Données (*D-*) ; 3. Données (*D+*) ; 4. Masse (*GND*)

Fig. 3. Formes des connecteurs USB[1]

Les adresses des périphériques USB sont codées sur 7 bits donc, en théorie, 128 périphériques (2^7) différents peuvent être connectés simultanément à un port de ce type. Les ports USB supportent le “Hot plug and play”. Ainsi, les périphériques peuvent être branchés sans éteindre l’ordinateur. Lors de la connexion du périphérique à l’hôte, ce dernier détecte l’ajout du nouvel élément grâce au changement de la tension entre les fils D+ et D-. A ce moment, l’ordinateur envoie un signal d’initialisation au périphérique pendant 10 ms, puis lui fournit du courant grâce aux fils GND et VBUS (jusqu’à 100mA). Le périphérique est alors alimenté en courant électrique et récupère temporairement l’adresse par défaut (l’adresse 0). L’étape suivante consiste à lui fournir son adresse définitive (c’est la procédure d’énumération). Pour cela, l’ordinateur interroge les périphériques déjà branchés pour connaître la leur et en attribue une au nouveau, qui en retour s’identifie. L’hôte, disposant de toutes les caractéristiques nécessaires, est alors en mesure de charger le pilote (“driver”) approprié.

Le bus USB est utilisé pour relier à l’ordinateur des périphériques de tous types, comme la souris, des lecteurs de cartes, des appareils photos et des disques. Il remplace la liaison série RS-232. Comme les transmissions sont généralement gérées par un logiciel (il n’y pas de hardware intelligent), les performances sont altérées lorsque le processeurs est très occupé (mais il ne coûte pas cher à installer).

[1] Le connecteur est celui utilisé sur les consoles de jeux GAMEBOY des années 80.

2.1.5.- FireWire (IEEE1394)

Le bus **IEEE 1394**[2] (du n° de la norme qui décrit le protocole) a été mis au point en 1995 afin de fournir un système d'interconnexion permettant de faire circuler des données à haute vitesse en temps réel (c-à-d en mode synchrone - cf. cinéma).
Il permet de connecter des périphériques à très haut débit (comme des caméras numériques et des disques durs). Les connecteurs et câbles FireWire sont repérables grâce à leur forme, ainsi qu'à la présence d'un logo en étoile à trois branches :

Fig. 4. Connecteurs FireWire et leur logo

Une liaison FireWire est basée sur un bus série qui fonctionne grâce à un "hardware" spécifique (donc coûteux) et utilise un câble blindé composé de six fils (deux paires pour les données et pour l'horloge, et deux fils pour l'alimentation électrique). Elle permet d'atteindre des débits de 0.1 à 3.2 Gbits/s. Le protocole permet de "chaîner" en grand nombre de périphériques (max. théorique 2^16 mais en pratique 3 ou 4). Le FireWire permet d'obtenir une vitesse de transmission garantie même lorsque le processeur principal est occupé (mode isochrone) et est donc spécialement bien adapté à la gestion des périphériques "multimédia" (vidéo, TV, etc...).

2.1.6.- Bluetooth (IEEE802)

La technologie **Bluetooth**[3] est devenue une norme officielle en juillet 1999. Il s'agit d'une technologie utilisant des émetteurs/récepteurs radio de faible portée qui permet de faire dialoguer entre eux plusieurs appareils. Bluetooth permet de transmettre des données ou de la voix entre des équipements possédant une interface radio. Bluetooth permet d'obtenir des débits de l'ordre de 1 à 3 Mbits/s avec une portée d'une dizaine de mètres environ. La technologie sans fil Bluetooth fonctionne partout dans le monde sur la bande de fréquences de 2,4 GHz (proche de celles utilisées dans les fours à micro-onde). Bluetooth élimine les traditionnels enchevêtrements de câbles et est surtout adapté à relier entre eux les ordinateurs, des téléphones mobi-

[2] La société *Apple* lui a donné le nom commercial « **Firewire** », qui est devenu le plus usité. Sony lui a également donné le nom commercial de **i.Link**, tandis que *Texas Instrument* lui a préféré le nom de *Lynx*.

[3] Originairement mise au point par la société suédoise *Ericsson* en 1994, son nom fait référence au roi danois Harald II (910-986), surnommé Blåtand (« à la dent bleue »), à qui on attribue l'unification des pays nordiques.

les (équipés ou non d'un appareil photo intégré), des imprimantes et des appareils de prises de vues numériques ainsi que les casques, claviers et souris. Les puces BlueTooth peuvent être fabriquées à faible coût et devraient, à terme, être présentes dans des tas d'appareils.

En utilisation normale, un périphérique fonctionne en « mode passif », c'est-à-dire qu'il est à l'écoute du réseau. L'établissement de la connexion commence par une phase appelée « phase d'inquisition » pendant laquelle le périphérique maître envoie une requête à tous les périphériques présents dans la zone de portée, appelés points d'accès. Tous les périphériques recevant la requête répondent avec leur adresse. Le périphérique maître choisit une adresse et se synchronise avec le point d'accès. Un lien s'établit ensuite avec le point d'accès, permettant au périphérique maître d'entamer une phase de découverte des **services du point d'accès**, selon un protocole appelé SDP (Service Discovery Protocol). A l'issue de cette phase de découverte de services, le périphérique maître est en mesure de créer un canal de communication avec le point d'accès en utilisant le protocole spécialisé (L2CAP). Il se peut que le point d'accès intègre un mécanisme de sécurité permettant de restreindre l'accès aux seuls utilisateurs autorisés et disposant d'un « code PIN » (Personal Identification Number).

Le standard Bluetooth définit un certain nombre de profils d'application (*Bluetooth profiles*), permettant de définir le type de services offerts par un périphérique Bluetooth. Chaque périphérique peut ainsi supporter plusieurs profils. Voici une liste des principaux profils Bluetooth :

- Advanced Audio Distribution Profile (A2DP) : profil de distribution audio avancée
- Audio Video Remote Control Profile (AVRCP) : profil de télécommande multimédia
- Basic Imaging Profile (BIP) : profil d'infographie basique
- Basic Printing Profile (BPP) : profil d'impression basique
- Cordless Telephony Profile (CTP) : profil de téléphonie sans fil
- Dial-up Networking Profile (DUNP) : profil d'accès réseau à distance
- Fax Profile (FAX) : profil de télécopieur
- File Transfer Profile (FTP) : profil de transfert de fichiers
- Generic Access Profile (GAP) : profil d'accès générique
- Generic Object Exchange Profile (GOEP) : profil d'échange d'objets
- Hardcopy Cable Replacement Profile (HCRP) : profil de remplacement de copie papier
- Hands-Free Profile (HFP) : profil mains libres
- Human Interface Device Profile (HID) : profil d'interface homme-machine
- Headset Profile (HSP) : profil d'oreillette
- Intercom Profile (IP) : profil d'intercom (talkie-walkie)
- LAN Access Profile (LAP) : profil d'accès au réseau
- Object Push Profile (OPP) : profil d'envoi de fichiers
- Personal Area Networking Profile (PAN) : profil de réseau personnel
- SIM Access Profile (SAP) : profil d'accès à une carte SIM
- Service Discovery Application Profile (SDAP) : profil de découverte d'applications
- Synchronization Profile (SP) : profil de synchronisation avec un gestionnaire d'informations personnelles (appelé *PIM* pour *Personal Information Manager*).
- Serial Port Profile (SPP) : profil de port série

Plus de détail en http://www.commentcamarche.net/bluetooth/bluetooth-fonctionnement.php3

2.2.- Les réseaux locaux de communication

2.2.1.- Les supports physiques

Nous décrivons ci-après les quatre types de supports les plus souvent utilisés pour relier des machines au sein d'un réseau. Il s'agit toujours de liaisons "à un seul fil" (plus une masse) sur lesquelles les données sont sérialisées. Lorsque les distances sont longues, il serait en effet peu rentable de transmettre les informations en parallèle. Le choix entre les différentes solutions possibles est habituellement guidé par les critères suivants :
-la vitesse maximale de transmission,
-la distance maximale de transmission (sans accessoire supplémentaire),
-l'immunité aux parasites et aux interférences,
-le prix.

Il existe d'autres supports (liaisons par ondes ultracourtes, par satellites, par signal infrarouge, etc…) mais ils ne sont habituellement pas utilisés pour établir des réseaux locaux.

1.- Câble téléphonique (twisted pair)
Il s'agit du support le moins coûteux. Il permet des liaisons à faible distance et est très simple à installer. Il ne permet malheureusement pas d'atteindre des vitesses de transmission extrêmement rapides et est assez sensible aux parasites.

2.- Câble coaxial
Il s'agit d'un câble semblable à celui utilisé pour la distribution de la télévision par câble. Il est formé d'un conducteur central isolé, entouré d'une gaine conductrice. Il permet d'atteindre des vitesses de transmission élevées (16 millions de bits par seconde et même plus). Son coût d'installation est raisonnable. Il est habituellement choisi pour implémenter un réseau local dans un bâtiment.

3.- Fibre optique
La lumière (plutôt que des impulsions électriques) est utilisée pour transmettre l'information dans des câbles optiques. Cette technique, plus complexe, permet d'atteindre des vitesses de transmission extrêmement élevées (plusieurs Gbits/sec) et présente une très bonne immunité aux parasites. Actuellement, le prix des interfaces et des connecteurs pour fibres optiques est encore relativement élevé et cette solution est habituellement réservée aux liaisons à longue distance, éventuellement multiplexées.

4.- Ondes radio (WiFi IEEE802-11) 
Le WiFi est un réseau en bus à transmission par paquet qui permet de véhiculer l'Internet par liaison radio entre une ou plusieurs bornes émettrices (AP - Acces Point) et un ordinateur qui dispose d'une carte client (station). Chaque borne diffuse

régulièrement (à raison d'un envoi toutes les 0.1 secondes environ) un message donnant des informations sur ses caractéristiques. Le client dialogue alors avec chacune des bornes pour évaluer la qualité des transmissions et choisir dynamiquement le point d'accès qui donnera le meilleur débit. Lorsque le réseau WiFi est choisi et que la liaison est établie, le client est relié à l'Internet et, s'il se déplace, il peut sauter d'une borne à l'autre tout en gardant sa liaison active. On peut introduire des mots de passe pour limiter l'accès d'un réseau WiFi aux machines clientes autorisées.

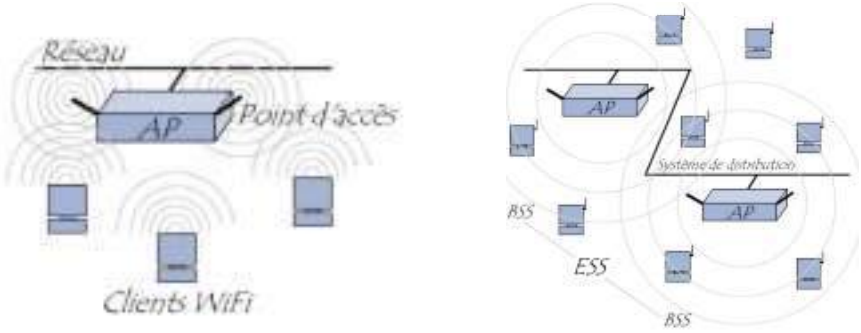


Fig. 5. Topologie d'un réseau WiFi à point d'accès (AP) unique ou multiple

La norme 802.11g la plus souvent utilisée offre un débit de 54 Mbps théoriques, (30 Mbps réels) sur la bande de fréquence des 2.4 GHz. Elle est cependant compatible avec la norme plus ancienne 802.11b (11Mbits/s). (Plus d'information en http://www.commentcamarche.net/wifi/wifiintro.php3).

2.2.2.- Topologies les plus courantes

LE RESEAU EN ETOILE (ET EN GRAPHE)

Chaque station est connectée individuellement à un poste central dédicacé à la gestion du réseau. Ce poste central s'occupe du routage des messages en mettant en communication les postes concernés par commutation des lignes en fonction des adresses demandées. Ce fonctionnement est proche de celui des lignes téléphoniques. Actuellement, la plupart de ces dispositifs groupent les informations en paquets qui sont envoyés à destination sur des lignes multiplexées selon des méthodes compliquées ("Packet Switching"). La gestion des réseaux (qui ont une structure en graphe) est assurée par des logiciels de communication spécialisés. En pratique, chaque poste se relie à un centre PBX (Private Branch eXchange) qui sert de porte d'entrée sur le réseau. En Belgique, les PTT utilise le standard "X-25" pour supporter des liaisons en paquets entre différents ordinateurs.

LE RESEAU EN BUS

Les ordinateurs qui constituent des stations de travail sont reliés à un câble unique banalisé (le bus) selon une technique qui permet de connecter et déconnecter à volonté chaque station sans interrompre les transmissions. Le bus est banalisé, il n'y a pas d'arbitre. Le protocole habituellement utilisé repose sur le concept appelé **CSMA/CD** ("Carrier Sense Multiple Acces with Collision Detection"). Si une station veut parler, elle attend que le bus soit libre ("carrier sensing") et envoie son message qui contient les clés-adresses identifiant l'émetteur et le receveur (codées sur 48

bits, un peu comme un numéro de téléphone). Chaque station est toujours à l'écoute mais, puisque chaque message est précédé par l'adresse du ou des receveurs, dès qu'une station sait qu'elle n'est pas concernée, elle cesse de décoder le message et ne reprend son attention qu'au début du message suivant.
Pour éviter la monopolisation du bus, les transferts sont découpés en paquets relativement petits (de 500 an 1500 octets). La vitesse effective des transmissions dépend de la charge du réseau et atteint typiquement un million de bytes par seconde.
Si plusieurs stations sont en attente, il peut y avoir collision, c'est-à-dire que plusieurs stations cherchent à parler exactement en même temps lorsque le bus se libère. La technique utilisée pour détecter une collision consiste à vérifier très rapidement si la communication n'est pas brouillée (on relit ce que l'on envoie - "collision detection"). S'il y a erreur, cela indique généralement une collision. On tente alors de réémettre le message ("multiple access"), mais après un délai aléatoire différent pour chaque station, ce qui évite que des collisions se reproduisent systématiquement.
Ce principe est utilisé sur le réseau Ethernet, AppleTalk et WIFi (avec une légère variente : il y a accusé de reception). Plus d'informations en
http://www.commentcamarche.net/wifi/wifimac.php3.

LE RESEAU A JETONS (Token ring)

Le réseau à jetons utilise une topologie en anneau. Chaque station est connectée à seulement deux autres stations (la précédente et la suivante) et la connexion forme un anneau fermé. Une des stations joue le rôle d'arbitre et provoque continuellement l'émission d'un message (le jeton) vers la station qui la suit. Lorsqu'une station reçoit un jeton, trois cas peuvent se présenter.

• A- Le jeton contient un message non vide qui ne lui est pas destiné : la station réémet automatiquement ce message vers la station suivante.
• B- Le jeton contient un message destiné à la station concernée : la station lit et décode le message puis retransmet un jeton vide à la station suivante.
• C- La station désire émettre un message : dès qu'un jeton vide lui est transmis, elle place le message à transmettre dans le jeton et l'adresse au destinataire. Ce message est transmis de station en station jusqu'à la station réceptrice (cf. cas A).

L'arbitre surveille les transmissions et évite qu'un jeton non vide fasse plus d'un tour (ce qui signifie que l'adresse demandée n'existe pas et pourrait bloquer le réseau). Notons que dans un réseau en anneau, il faut que toutes les stations soient opérationnelles pour que le réseau fonctionne correctement et que, pour intercaler une nouvelle station, il est nécessaire d'interrompre momentanément les transmissions. Le Token Ring est en voie d'extinction.

3.- LA PROGRAMMATION DES INTERFACES

3.1.- Interface système

Certains périphériques sont suffisamment classiques pour que le constructeur prévoit leur gestion dans le système d'exploitation de son ordinateur. C'est le cas des lecteurs de disques, des imprimantes et des modems. La gestion de ces interfaces et des périphériques associés est assurée par le système d'exploitation qui utilise des petits programmes appelés "drivers" pour s'adapter aux interfaces.

3.2.- Interface particulière

Pour gérer une interface particulière il faut généralement charger un programme pilote ou “driver” fourni par le constructeur de l’interface et qui assure l’adaptation entre le système d’exploitation et les programmes de l’utilisateur. Les instructions disponibles dans les langages de haut niveau sont alors capables d’accéder à l’interface.

Une carte d'interface est généralement branchée sur le bus principal de l'ordinateur. Ce bus contient entre autres des fils d'adresses et des fils de données. Sur l'interface, un décodeur d'adresses réagit chaque fois que certaines adresses sont placées sur les fils d'adresses. Ces adresses doivent être uniques et sont spécifiques à l'interface (on peut les obtenir en consultant la documentation technique). La réaction de la carte dépend de l'adresse activée et de l'état de certaines autres lignes du bus. Elle peut, par exemple, lire ou écrire une donnée sur les fils de données, provoquer une opération particulière (basculer un signal, allumer une lampe, induire une interruption, etc…). Sur les très petits microcontrôleurs, il est encore possible d’agir directement sur les signaux des bus par programme. Par contre, pour les systèmes d’exploitation modernes (Windows XP, OS X, Linux, etc...) cela n’est plus possible et il faut donc toujours passer par le “driver”. Des informations sur la programmation d’un microcontrôleur sont disponibles en : http://www.ipnas.ulg.ac.be/garnir/avr/.

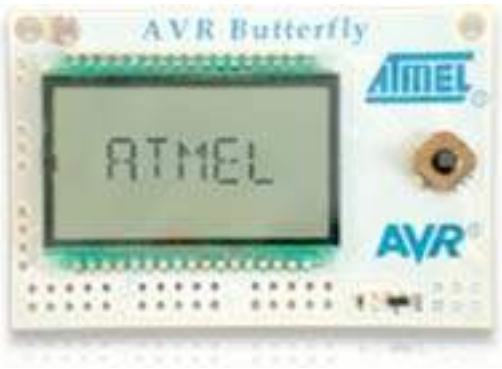


Fig. 6. Une carte microcontôleur avec affichage LCD

4.- LES INTERRUPTIONS

Une interruption est un événement (souvent extérieur à la machine) qui déroute brutalement l'activité du processeur vers des routines spécialisées qui "traitent l'interruption" avant de rendre le contrôle au processus original. La notion d'interruption a ouvert la voie aux programmes " temps réel" qui peuvent réagir immédiatement aux stimulations externes. La plupart des processeurs disposent des connexions leur permettant d'être déroutés par des interruptions. Le fonctionnement exact de ces processus est cependant différent d'un processeur à l'autre.

C H A P I T R E XIII

Les périphériques usuels

1.- LES PERIPHERIQUES DE STOCKAGE DES DONNEES

Ces périphériques sont indispensables pour garder les informations à long terme. Nous décrivons ci-après leur fonctionnement.

1.1.- Disque souple (floppy disk)

Un disque souple ou disquette se compose d'un disque mince de plastique recouvert sur les deux faces d'une fine pellicule de substance magnétisable. Le disque est protégé par une pochette percée de trous. Le trou central permet de mettre le disque en rotation tandis que le trou radial permet d'accéder aux surfaces du disque.

Fig. 1. Disques souples (ou disquettes)

Les lecteurs de disquettes souples utilisent une technique d'encodage magnétique. La surface du disque est recouverte d'une substance qui contient en suspension un très grand nombre de micro-particules magnétisables. Le disque est mis en rotation par un moteur. Les têtes de lecture sont fixées sur un support mobile capable de se déplacer radialement (fig 2). Chaque tête contient un électro-aimant qui est capable, en lecture, de mesurer les variations de flux engendrées par le défilement de la surface magnétique. En écriture, ces mêmes têtes induisent un champ local intense qui modifie de façon durable le sens de magnétisation des surfaces qui défilent sous elles.

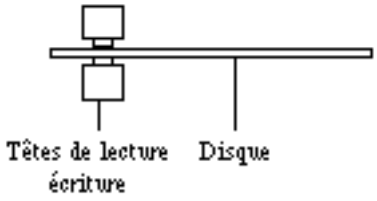


Fig. 2. Schéma de principe d'un lecteur de disquettes (vue latérale)

En pratique, toutes les informations sont codées sous forme d'une suite de 0 et de 1 correspondant aux deux sens de magnétisation. Un micro-processeur spécialisé assure le codage et le décodage des micro-variations de flux. Il contrôle aussi la rotation du moteur ainsi que le positionnement exact de la tête sur les différentes pistes concentriques.
La fréquence d'écriture sur les disquettes est de l'ordre de 200 à 300 kHz tandis que la vitesse de défilement est de ≈300cm/s. (En comparaison, les rubans de cassette audio défilent à 4.8 cm/s et ont une bande passante de ≈12KHz.)
Les lecteurs de disquettes sont disponibles chez tous les constructeurs d'ordinateurs. Les formats des disquettes sont les mêmes chez tous. Il existe deux formats courants.
(a) Les disquettes de 5"1/4 (≈13cm), sous pochette souple, qui tournent à 300 t/min et ont une capacité de 150 à 400 koctets.
(b) Les disquettes 3"5 (≈9 cm), sous pochette rigide, qui tournent à ≈500 t/min et ont une capacité de 400 à 1400 koctets. L'ancien format de 8" (≈20 cm) tend à disparaître. Bien que les caractéristiques mécaniques des lecteurs des différentes marques soient identiques, les méthodes d'encodage de l'information diffèrent d'un constructeur à l'autre. A cause de cela, il n'est pas toujours possible de transférer facilement des données entre les machines de marques et de modèles différents.

Signalons que les disquettes sont de moins en moins utilisées depuis l'apparition des mémoires flash (cf. § 1.3.).

1.2.- Disque dur

Les disques durs fonctionnent selon les mêmes principes que les disquettes cependant ils sont construits différemment. Les informations sont rangées sur plusieurs plateaux superposés tournant à très grande vitesse (3600 à 4260 tours/min). Ces plateaux sont rigides (d'où le nom "disque dur") et recouverts, sur les deux faces, d'une fine pellicule magnétique. Il y a deux têtes de lecture/écriture par plateau. Ces têtes sont placées aux extrémités d'un peigne mobile qui permet le déplacement radial des têtes sur toute la surface active des plateaux (fig. 3).

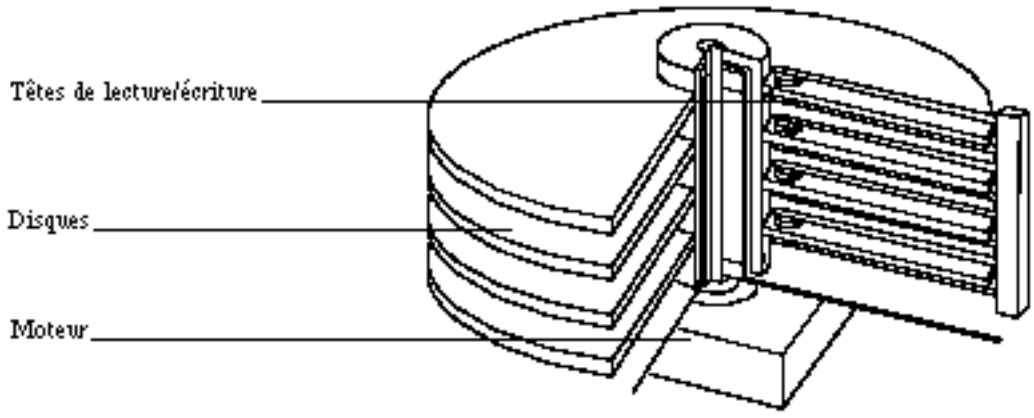


Fig. 3. Vue éclatée d'un disque dur

Pour limiter l'usure, les têtes ne touchent pas la surface des disques mais la survolent sur un coussin d'air extrêmement mince (de l'ordre de 0,25μm, soit 1/3 du diamètre d'une particule de fumée). Pour éviter tout incident, le dispositif est assemblé dans une pièce propre, exempte de toute poussière, et est livré enfermé dans un boîtier étanche. Grâce à la précision du mécanisme, il est possible d'utiliser un très grand nombre de pistes et de stocker une très grande quantité d'informations. La capacité des disques durs actuels va de 80 Moctets pour les plus petits (de la taille d'une boite d'allumettes à plus de 1 Teraoctets pour les gros disques utilisés dans les centres de calcul et les applications multimédia.
Une caractéristique importante des disques durs est liée à la vitesse de déplacement radial du support des têtes. Celle-ci est en rapport direct avec la vitesse d'accès à l'information. On parle souvent du "temps moyen d'accès" (access time) qui est le temps mort qui sépare la demande d'une information de son arrivée. On parle aussi du temps de déplacement des têtes (seek-time) qui correspond au temps moyen mis par le support des têtes pour passer d'une piste à l'autre. Il est généralement de l'ordre de 4 à 10 ms pour les disques durs utilisés sur les PC.

1.3.- Les sticks USB et les mémoires flash

Il s'agit de périphériques permettant de stocker l'information à long terme dans une mémoire à semi-conducteurs qui ne s'efface pas, même non alimentée. La technologie utilisée repose sur l'utilisation de mémoires "flash" qui se composent de petites cellules élémentaires qui se comportent chacune comme un petit condensateur qui est capable de conserver sa charge pendant une très longue période (on parle de 10 ans!). Chacune de ces cellules mémorise donc 1 bit d'information. Lors de la lecture, on mesure le champ induit par la charge de la cellule pour connaître l'état du bit. Pour écrire dans une mémoire flash, il faut au préalable "effacer" les cellules. Pour des raisons techniques elles doivent être effacées toutes en même temps et ensuite réécrites (d'où le vocable "flash" qui fait référence à cette opération). En pratique cependant, pour les mémoires de grande capacité, les fabricants partitionnent leurs circuits en plusieurs blocs et chacun des blocs peut être effacé et réécrit individuellement ce qui permet de mieux gérer les transferts.

Fig. 4. Différents formats de mémoires flash (dont une clé USB et un lecteur mp3)

Les mémoires flash se distinguent par leur capacité (de 16k à plusieurs Goctets) et leur vitesse de transfert (il est à noter que la vitesse d'écriture est beaucoup plus lente que la vitesse de lecture) et la fonctionnalité du circuit d'interfacage qui comporte souvent un micro-processeur spécialisé qui gère les processus de transfert

(par ex. une liaison USB ou un lecteur de cartes). Sur la plupart de ces mémoires, un interrupteur mécanique permet de protéger les données en interdisant l'écriture.

Les mémoires flash sont de plus en plus rapides et fiables (pas de pièce mobile). Leur capacité augmente sans cesse. Elles remplacent les disquettes et sont déjà utilisées pour stocker l'information dans toute une gamme de produits (GSM, appareil photo digital, baladeur, etc...). Elles pourront sans doute rapidement concurrencer les disques durs dans les ordinateurs.

1.4.- Disque optique[4]

L'utilisation de processus optiques permet d'augmenter la densité maximale d'informations que l'on peut enregistrer par unité de surface. La surface des disques optiques est aussi divisée en pistes concentriques mais on utilise des processus optiques pour lire les informations. C'est la fluctuation de la réflexion d'un mince faisceau laser sur la surface qui permet de relire l'information digitale (pas de réflexion : =0, réflexion importante : =1). Ces processus sont à la base des lecteurs de compacts disques digitaux (CD) utilisés dans le grand public.

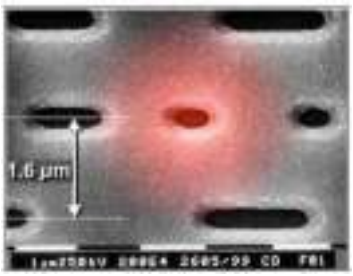


Fig. 5. Spot du laser sur la surface d'un CD

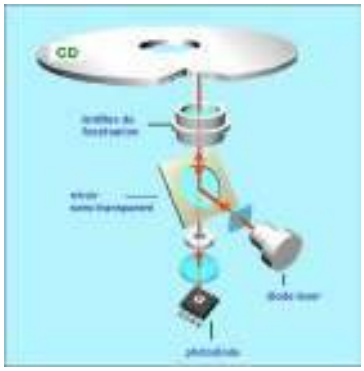


Fig. 6. Système optique de lecture par réflexion

[4] Plus d'information en http://www.ta-formation.com/acrobat-lib/optiq.pdf

1.4.1.- CD-ROM

Les CD-ROMs (Compact Disk - Read Only Memory) pour ordinateur ont exactement le même aspect que les CD musicaux. D'ailleurs, les lecteurs utilisés conjointement avec les ordinateurs peuvent lire indifféremment les CD-ROM et les CD musicaux. Les CD sont fabriqués industriellement par réplique à partir d'une matrice mère. L'original de cette matrice est obtenu par déformation, à l'aide d'un puissant laser pulsé, d'une feuille mince d'aluminium. (Les bosses engendrées par les chocs des photons diminuent la réflectivité de la surface.) La conception de la matrice est une opération coûteuse, mais ensuite, la fabrication en série d'un disque compact revient à moins de 0.1 €.

Fig. 7. CD-ROM

Les CD-ROMs d'ordinateur peuvent contenir 720 Moctets d'informations facilement accessibles (cf. les normes ISO et "High-Sierra"). Ils sont principalement utilisés pour distribuer des informations générales (dictionnaires, programmes d'ordinateurs, listes d'adresses ou de prix, images digitales, etc…). Il n'est cependant pas possible de changer leur contenu qui est définitivement figé et la conception d'une matrice est une opération coûteuse, lente et délicate.
Il est possible, pour un prix très faible, de recopier sur CD-ROM des images ou des films. Ces images peuvent ensuite être manipulées digitalement par un ordinateur ou simplement visualisées sur un écran de télévision.

1.4.2.- Les CD inscriptibles ou WORM

Les lecteurs WORM (pour Write Once, Read Many - écriture unique, lecture multiple) permettent d'écrire sur des disques optiques spéciaux en modifiant leur surface (comme dans les CD). On ne peut pas modifier ce qu'on a écrit mais l'information peut être relue aussi souvent qu'on le veut. Ce type de disque est principalement destiné à l'archivage. Sur les CD enregistrables les données ne sont plus stockées sous formes de creux mais enregistrées dans une couche de matériau photosensible. Lors de l'enregistrement sur un CD±R, les données sont inscrites par élévation de température (burning) à l'aide d'une impulsion laser qui échauffe le matériau organique (initialement transparent) au-delà de sa température critique de polymérisation. Il devient alors opaque de manière irréversible : on a créé l'équivalent d'un creux ou « pit ». La puissance du laser est de l'ordre de 10 mW et peut porter très rapidement le point de focalisation dans la résine photosensible à une température

d' environ 250 °C. Lors de la lecture, l'intensité reçue est varie donc en fonction de la transparence de la couche photosensible.

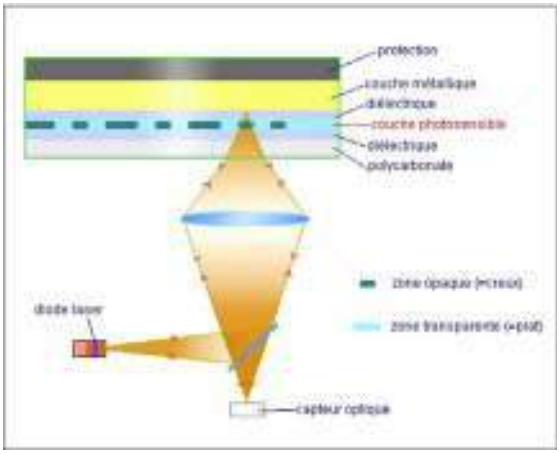


Fig. 8. Dispositif de lecture d'un CD±R

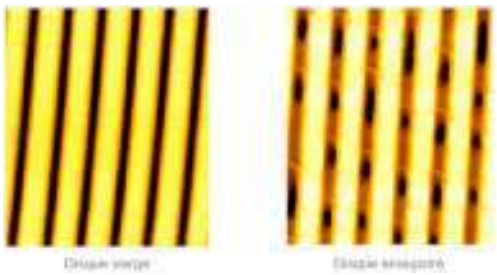


Fig. 9. Surface d'un CD±R avant et après gravure

Les CD±R disposent de pistes pré-tracées qui seront modulées par le laser d'écriture (d'où les deux standards + et - pour lesquels la prégravure est légèrement différente).

1.4.3.- Les CD et DVD réinscriptibles ou CD±RW et DVD±RW

Pour les disques réinscriptibles (±RW), le changement d'état du matériau photosensible est réversible : dans son état original (disque vierge), la couche d'enregistrement a une structure polycristalline transparente et l'écriture se fait comme pour un disque ±R. Pour effacer, on utilise une puissance laser plus faible mais continue (de l'ordre de 5 mW) qui chauffe légèrement (200 degrés) tout le volume du polymère qui devient liquide et qui se refroidit dans l'état cristallin (et donc transparent).

Ce type de disque réfléchit moins bien la lumière qu'un disque classique et ne peut donc être lu que par des lecteurs « Multiread » adaptés.

Un CD contient environ 720 Moctets d'informations. Un DVD utilise un encodage plus dense et peut stocker de 4,7 à 9,4 (double couche) Goctets. Cette capacité va encore augmenter dans un proche avenir.

2.- LES IMPRIMANTES

Parmi les périphériques habituels des ordinateurs, on trouve généralement au moins une imprimante destinée à produire des documents permanents qui pourront être consultés à loisir. Différentes techniques d'impression sont utilisées. Nous allons en décrire quelques unes.
(cf. aussi http://www.commentcamarche.net/pc/imprimante.php3)

2.1.- Imprimante à aiguilles ou matrice de points (“dot matrix”)

Une imprimante à aiguilles comporte une tête d'impression mobile qui contient un certain nombre d'aiguilles alignées verticalement. Chaque aiguille peut être propulsée vers l'avant par un électro–aimant. Chaque fois qu'une aiguille est mise en mouvement, elle comprime le ruban encreur sur le papier, ce qui imprime une petite tache noire ("dot"). La tête se déplace à vitesse constante de gauche à droite (et de droite à gauche pour les imprimantes bidirectionnelles) et imprime une ligne complète à chaque passage. Après chaque balayage, le papier est avancé de la distance d'un interligne. Une électronique adéquate active très rapidement les électro-aimants pour former une image à partir d'un très grand nombre de petites taches contiguës. La figure 10 présente la formation de la lettre "A" avec une imprimante disposant d'une tête à 7 aiguilles.

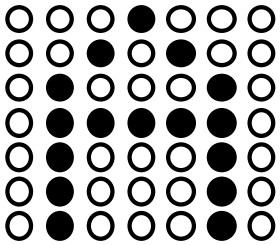


Fig. 10. Exemple d'impression par aiguilles

Ce type d'imprimante peut reproduire non seulement du texte mais aussi des dessins avec une très faible résolution.

Fig. 11. Imprimante à aiguilles

La vitesse d'impression est directement liée à la vitesse de frappe des aiguilles et peut atteindre de 80 à 200 caractères par seconde environ (soit 200 à 300 points/s). Plus la tête comporte d'aiguilles, plus la précision du tracé sera bonne. On trouve

actuellement des imprimantes dont la tête contient 24 aiguilles et qui ont une qualité d'impression très proche de celle des machines à écrire classiques (on les appelle imprimantes NLQ pour "Near Letter Quality").

2.2.- Imprimante "LASER"

Le schéma de principe de l'imprimante laser est présenté dans la figure 12. Le fonctionnement d'une imprimante laser est identique à celui d'une photocopieuse dans laquelle l'image reproduite serait une image synthétique créée par le balayage rapide d'un faisceau laser.

Le fonctionnement est le suivant.
1.- Un tambour rotatif recouvert de sélénium reçoit, par induction, une charge électrique uniforme lorsqu'il passe devant un balai porté à un potentiel électrique très élevé.
2.- Les régions qui sont éclairées par le faisceau du laser perdent leur charge par effet photo-électrique (le sélénium a un seuil photo-électrique très bas). L'intensité du laser est contrôlée par un dispositif électromécanique complexe qui reproduit séquentiellement l'image du document sur le tambour sous forme d'un très grand nombre de lignes (≈12/mm) longitudinales grâce notamment à un miroir tournant.
3.- Le tambour passe au-dessus d'un bac contenant une poudre très fine qui est attirée par effet électrostatique et vient se coller sur les zones chargées du tambour (c'est-à-dire celles qui n'ont pas été illuminées par le laser).
4.- Le tambour est pressé sur le papier et la poudre se décolle du tambour pour s'incruster sur le papier.
5.- Le papier passe ensuite dans un four qui fait fondre la colle que contient la poudre, ce qui fixe définitivement l'image (dont les zones noires correspondent aux zones non illuminées du tambour).

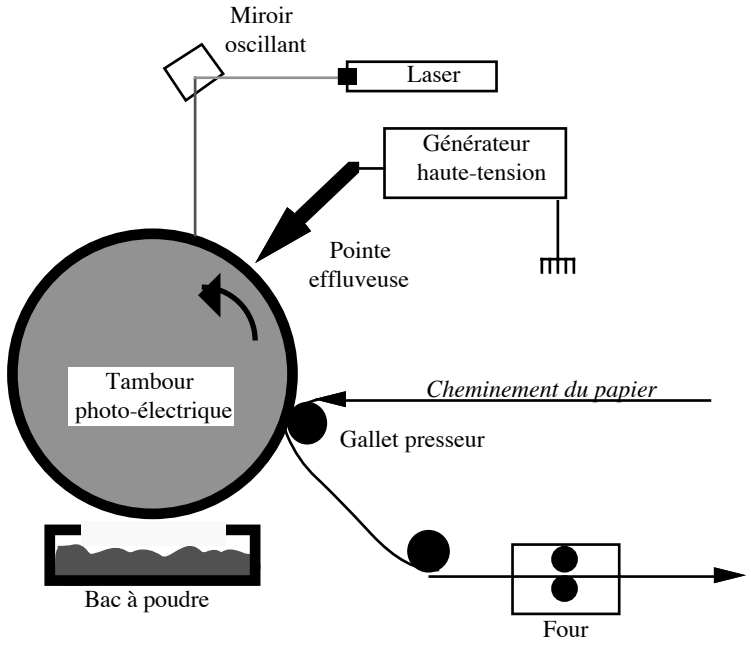


Fig. 12. Schéma de principe d'une imprimante laser

Fig. 13. Imprimante laser

La qualité d'une imprimante laser dépend du type de mécanisme qu'elle contient et des algorithmes qu'elle utilise pour décrire les images à reproduire. La résolution est définie par le nombre de points qui peuvent être imprimés sur une longueur d'un pouce.

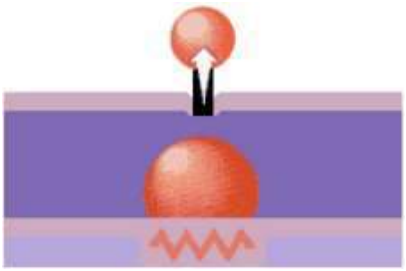
Pour les imprimantes destinées aux micro-ordinateurs, la résolution est habituellement de 300 ou 600 points/pouces. La vitesse d'impression peut atteindre 6 à 50 pages par minute.

Le même procédé d'impression est aussi utilisé sur les grosses imprimantes professionnelles qui permettent d'imprimer avec une qualité moyenne plusieurs centaines de pages par minute.

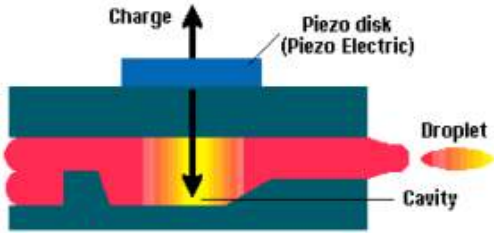
2.3.- Imprimante à jet d'encre

Dans les imprimantes à jet d'encre, les caractères sont formés par la juxtaposition d'un très grand nombre de petites taches produites par projection sur le papier de très fines gouttelettes d'encre. Le mécanisme de projection est composé de plusieurs gicleurs extrêmement fins montés verticalement sur une tête d'impression mobile qui se déplace de gauche à droite lors de l'impression (même principe que les imprimantes à aiguilles). Les gouttelettes sont projetées par la surpression induite par une mini-résistance, placée dans le canal d'amenée de chaque gicleur, qui, chaque fois qu'elle est alimentée, échauffe l'encre jusqu'à ébullition (fig. 14) ou par effet piézo-électrique (déformation d'un cristal par un champ électrique)(fig. 15).

Sur les imprimantes Hewlett-Packard DeskJet, la tête d'impression comporte 50 gicleurs séparés de 80μm (300 dpi) qui peuvent chacun projeter au maximum 3600 gouttelettes de 130 pl (pico litre!) par seconde. Lorsque les gicleurs sont alimentés par des encres de différentes couleurs, il est possible de réaliser des impressions en couleurs.



La résistance échauffe l'encre qui, en se vaporisant, projette une goutte vers l'avant.
Fig. 14. Buse d'injection d'une imprimante à jet d'encre (thermique)



En se déformant sous l'effet de tensions, le disque comprime le canal ce qui éjecte une goutte.
Fig. 15. Buse d'injection d'une imprimante à jet d'encre (piézo-électrique)

Les imprimantes à jet d'encre sont silencieuses, relativement bon marché et permettent d'obtenir des résultats proches de ceux des imprimantes laser. Elles sont cependant très dépendantes de la qualité de l'encre et du papier qui ont une influence directe sur la qualité d'impression. La vitesse d'impression est de 2 à 5 pages par minute.

2.5.- Imprimante à papier thermique

Ces imprimantes utilisent du papier spécial qui noircit à la chaleur. Elles fonctionnent selon le même principe que dans les imprimantes à aiguilles mais la tête d'impression se compose de plusieurs mini-résistances dont l'échauffement fait noircir le papier chaque fois que le courant circule dans ces résistances.
La technique d'écriture est la suivante. La surface du papier est recouverte d'une multitude de micro-bulles dont la surface est blanche et le contenu noir. L'échauffement fait éclater la bulle qui change de couleur.
Les imprimantes thermiques sont simples à construire, mécaniquement robustes et très silencieuses. Cependant le papier thermique est relativement coûteux et a tendance à noircir spontanément avec le temps. Les imprimantes thermiques ne conviennent donc pas pour réaliser des documents de qualité. Notons que ce type d'imprimante est souvent proposé dans les appareils FAX (télécopie de documents par téléphone).

2.6.- Les photocomposeuses digitales

Lorsque l'on désire obtenir une qualité d'impression très élevée, on peut recourir à des dispositifs professionnels d'impression comme les photocomposeuses digitales. La résolution d'une photocomposeuse digitale peut atteindre 2400 à 3200 pixels/

pouce (limite de résolution de l'oeil). La plupart de ces appareils acceptent des fichiers de descriptions d'images écrits en PostScript. Les impressions en couleurs utilisent habituellement la technique de la quadrichromie qui consiste à réaliser quatre transparents : un bleu, un jaune, un rouge et un noir (pour les contours et les inscriptions) qui serviront à imprimer les quatre couches de couleurs successives qui permettent de réaliser sur papier des images couleurs. Actuellement, les logiciels professionnels réalisent automatiquement la séparation des couleurs et contrôlent tous les stades de l'impression des clichés. Le prix de ces machines de haut de gamme peut atteindre plusieurs centaines de milliers d'€.

SpeedMaster 74 DI d'Heidelberg

Fig. 16. La photocomposeuse digitale

2.7.- Les tables traçantes

Une table traçante permet de déplacer avec précision une plume sur la surface du papier. L'ordinateur commande les déplacements de la plume et construit l'image désirée par une suite de déplacements. Certaines tables traçantes permettent de réaliser des dessins de grande taille. Ce type d'imprimante convient surtout à la réalisation de dessins techniques (plans et cartes) ou de schémas électriques.
Les tables traçantes ne permettent pas facilement de dessiner des zones uniformes (aplats) et sont assez coûteuses à l'achat.

Fig. 17. La table traçante HP 1050c

3.- DISPOSITIFS DE POINTAGE

Pour faciliter le dialogue entre l'utilisateur et la machine, on a recours à divers dispositifs de pointage qui permettent de commander directement l'ordinateur sans utiliser

le clavier. Le plus célèbre est probablement la "souris" que nous allons décrire ci-après avec quelques autres dispositifs.

3.1.- Souris

Une souris est un dispositif de pointage qui commande le déplacement sur l'écran d'un pointeur (habituellement symbolisé par une petite flèche). La souris est devenue indispensable au bon fonctionnement des ordinateurs actuels. Son invention remonte à 1970. Les souris sont généralement dotées d'un ou de plusieurs boutons que l'utilisateur peut presser pour demander une action (on utilise le mot "cliquer"). La souris permet d'amener le pointeur dans des zones particulières de l'écran (menu, bouton, fenêtre etc…) qui réagissent chacune d'une façon particulière aux déplacements et aux clics. L'interface utilisateur a évolué vers une utilisation systématique de la souris qui remplace un grand nombre de commandes clavier (cf. les environnements Macintosh et Windows).

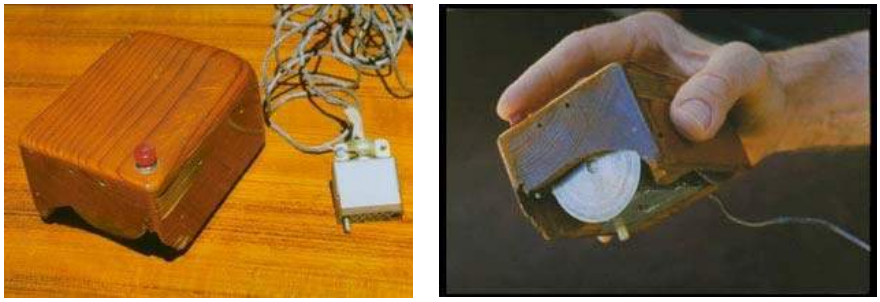


Fig. 18. La première souris d'ordinateur
inventée en 1970 au laboratoire PARC par Douglas Engelbart,
dont la licence a été achetée par Apple dans les années 80.
(http://en.wikipedia.org/wiki/Douglas_Engelbart)

Les premières souris fonctionnaient à partir de deux codeurs angulaires qui suivent les mouvements de rotation d'une grosse bille que l'on déplace sur une surface rugueuse (fig. 19).

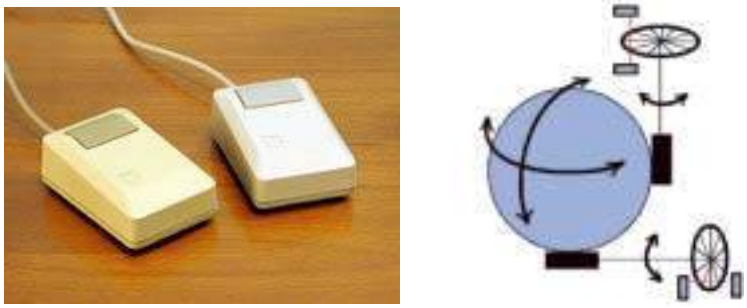


Fig. 19. Une souris à un bouton et son mode de fonctionnement

Les souris plus récentes reposent sur un capteur optique. Elles sont équipées d'une mini-caméra CMOS et d'une diode rouge ou d'un laser (il n'y a plus de boule). La surface éclairée est filmée par la caméra plusieurs milliers de fois par seconde et un processeur compare les images pour détecter tout déplacement. Ce système est

d'une très grande précision, fonctionne sans à-coups et ce quelle que soit la surface (sauf sur un miroir ou une surface lignée).

Fig. 20. Une souris optique à transmission sans fil

De plus en plus de souris sont équipées de plusieurs boutons (jusqu'à 5) et d'une molette ou d'un curseur. Ceci permet de faire défiler des pages tout en permettant à l'utilisateur de déplacer le curseur sur l'écran.

3.2.- Tablette graphique

Une tablette graphique permet d'enregistrer les mouvements d'un "crayon" spécial que l'on déplace sur sa surface. Les tablettes graphiques permettent par exemple de "lire" électroniquement l'écriture manuelle ou de reproduire le dessin que l'on trace sur sa surface. Une application classique des tablettes graphiques consiste à enregistrer les coordonnées relatives de divers points que l'on repère sur un document (carte ou plan) placé sur la tablette. Elles sont d'usage courant en cartographie.

Fig. 21. Une tablette graphique

3.3.- Crayon optique

Un crayon optique permet de désigner une zone particulière de l'écran en positionnant dessus l'extrémité active du crayon. En réalité, le crayon n'écrit pas, il contient un capteur de lumière qui détecte le passage du spot d'écriture sur l'écran et retransmet les coordonnées de ce point vers l'ordinateur. Autrefois concurrent de la souris, le crayon optique tend à disparaître car sa précision est faible et son utilisation prolongée est inconfortable.

3.4.- Ecran tactile

Les écrans tactiles réagissent au toucher. L'usager désigne une zone de l'écran en plaçant son doigt dessus. Deux techniques de détection sont utilisées. La première consiste à placer devant l'écran un dispositif transparent, sensible à la pression, qui permet de retrouver les coordonnées approximatives de l'endroit qui subit un écra-

sement. L'autre méthode consiste à entourer l'écran d'un grand nombre de diodes émettrices (LED) couplées à des détecteurs leur faisant face. Le fait d'avancer le doigt dans la zone active masque certains détecteurs et permet de déterminer les coordonnées de l'obstacle.
Les écrans tactiles ont un certain succès dans les milieux industriels car ils sont d'un usage évident et permettent d'activer facilement certaines opérations sans avoir recours à des dispositifs électromécaniques qui peuvent s'encrasser (comme les claviers ou les souris).
On constate un regain d'intérêt pour ce genre d'écran dans des petits appareils portables (PDA - Personal Digital Assistant) qui fonctionnent comme des blocs-notes électroniques intelligents. Des logiciels de reconnaissance de forme permettent de décoder l'écriture manuelle et d'interpréter les commandes indiquées par la simple pression d'un stylet sur l'écran.

4.- **LES SCANNERS** ou digitiseurs d'images

Un scanner (dans le milieu informatique) est un appareil qui permet de digitaliser un document. Il fonctionne à l'inverse d'une imprimante et permet de récupérer à l'écran une image électronique qui peut ensuite être manipulée par des logiciels spécialisés.

Fig. 22. Exemple d'une image digitalisée à faible résolution (N/B 72 pixels/pouce)

Il en existe deux familles, les scanners "à plat" ("flat-bed") et les scanners rotatifs ("drum"). La résolution des premiers est moins bonne mais leur usage est plus simple (et leur coût est moins élevé). Le scanner à plat utilise habituellement un détecteur CCD (Charge Coupled Device) comme celui d'une caméra vidéo. Le document est enregistré en mesurant la lumière réfléchie vers le détecteur mobile qui analyse le document de haut en bas (un peu comme dans une photocopieuse). Le scanner à tambour rotatif utilise comme détecteur un tube photomultiplicateur très sensible qui analyse la surface du document qui tourne devant lui en se déplaçant lentement de droite à gauche.

Les images, une fois digitalisées, peuvent être traitées par les programmes de traitement d'images (digital darkroom) ou des programmes particuliers comme les OCRs (Optical Character Recognition) qui "relisent" un texte.
Les scanners sont beaucoup utilisés dans les officines de publication assistée par ordinateur ("desktop publishing").

La résolution d'un scanner "flat bed" est de ≈300 à 400 pixels/pouce tandis que la résolution d'un scanner à tambour peut atteindre 3000 pixels/pouce. (Dans ces conditions, une image couleur de taille A4 se traduit en un fichier de ≈3 Goctets, qui sature les capacités des PC actuels.)
Tous les dispositifs FAX utilisent un scanner (résolution 200 pixels/pouce) pour enregistrer le document à transmettre.

Fig. 23. Scanner à Plat Scanner Vertical

Plus d'informations en http://www.library.cornell.edu/preservation/tutorial-french/technical/technicalB-03.html

5.- LES MODEMS

5.1.- Les modems classiques

Un modem est un appareil qui permet de transmettre des informations digitales sur les lignes du réseau téléphonique (fig. 24). Le nom vient de la contraction de MODulateur DEModulateur.

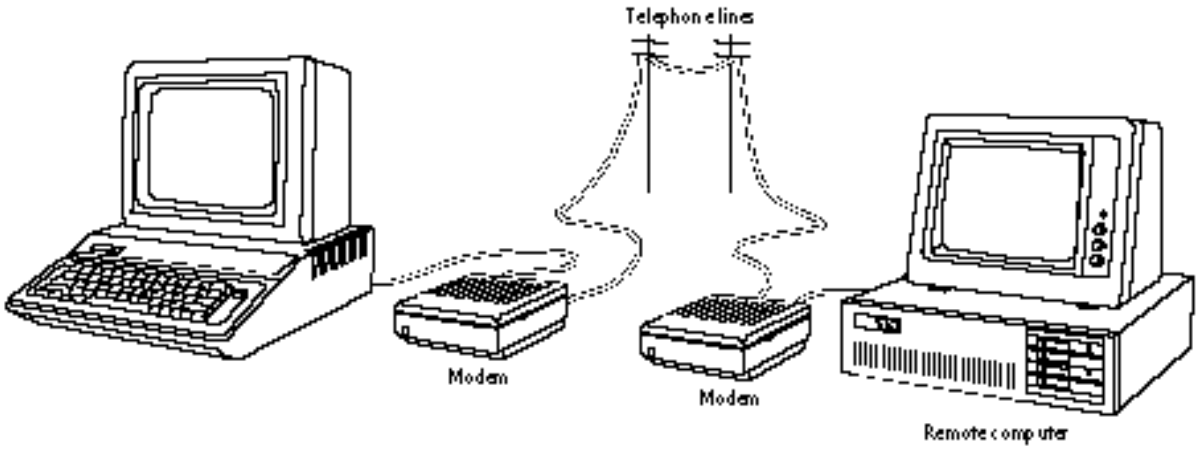


Fig. 24. Principe de la communication par modem

Le téléphone n'est prévu que pour transmettre la voix humaine (gamme de fréquence de 100 à 5000 Hz environ). Pour faire circuler sur des lignes téléphoniques des messages digitaux, il faut les encoder sous une forme qui peut transiter sans encombre sur les lignes téléphoniques en simulant la voix humaine. En pratique, on utilise une suite de sifflements dont les fréquences varient continuellement. A l'arrivée, ces signaux sont décodés pour reconstruire les données originales.

Le modulateur émet un son continu sur deux fréquences fixes (par exemple 1180Hz/980Hz (émetteur principal) et 1850Hz/1650Hz (émetteur secondaire) - ces fréquences sont bien tolérées par les installations des PTT…). Une des fréquences correspond à 0 et l'autre à 1. L'écoute du signal permet de reconstituer, à l'autre bout de la ligne, la succession des 0 et des 1 correspondant par exemple à une transmission selon le protocole RS232.

Lorsque la transmission est possible en même temps dans les deux sens, on parle de full-duplex. Si la transmission est bidirectionnelle mais qu'un seul modulateur à la fois peut émettre, on parle de half-duplex. Si la transmission se fait dans un seul sens, elle est dite simplex.
Les FAX utilisent des modems pour transmettre leurs documents à la vitesse moyenne de 9600 Bauds.

Les vitesses maximales de transfert d'informations sont, en principe, limitées par la faible bande passante des lignes téléphoniques. Cependant, l'ingéniosité des concepteurs de modems n'a pas de limite et les vitesses effectives de transmission ne cessent d'augmenter. Il est actuellement possible de transmettre plus de 5000 octets par seconde sur des lignes classiques. Pour atteindre des vitesses plus élevées, on peut utiliser des modems spéciaux (adsl) ou louer des lignes particulières (très chères). Les vitesses peuvent alors atteindre des millions de bits/s dans les meilleures cas.

5.2.- L'ADSL (Asymmetric Digital Subscriber Line)

Un modem ADSL permet de faire coexister sur une même ligne un canal descendant (downstream) de haut débit, un canal montant (upstream) moyen débit ainsi qu'un canal de téléphonie classique (POTS : Plain Old Telephone Service). L'ADSL permet de recycler les vieux réseaux de téléphonie fixe qui utilisent un câblage filaire composé de multiples paires torsadées (une par ligne ; plus de 800 millions de connexions dans le monde!) en ajoutant un équipement au central téléphonique ainsi qu'une petite installation chez l'utilisateur.

Les services téléphoniques traditionnels fonctionnent dans une gamme de fréquence très réduite (de 300 à 5k Hz). Or, on s'est aperçu que les câbles reliant les centraux téléphoniques aux utilisateurs peuvent véhiculer des fréquences plus élevées allant jusqu'à plusieurs centaines de kHz.

L'ADSL utilise le spectre de fréquence allant jusque 1,104 MHz qui est divisé en 256 sous-canaux distincts espacés de 4,3125 kHz. Les sous-canaux (1 à 6) de basse fréquence sont laissés libres et véhiculent la téléphonie analogique (POTS), ce qui assure la compatibilité avec la téléphonie classique.

On s'est aperçu qu'il était possible de transmettre les données plus rapidement d'un central vers un utilisateur que l'inverse. L'idée est donc d'utiliser un système asymé-

trique, en imposant un débit plus faible sur des fréquences plus basses de l'abonné vers le central (qui peut alors utiliser des puissances d'émission plus faibles). Les sous-canaux 7 à 31 sont donc alloués au flux montant (upstream - de l'utilisateur vers le central). Les sous-canaux 33 à 256 sont utilisés pour les flux descendants (downstream -vers l'utilisateur)[5] car l'équipement installé au central peut émettre avec une puissance très élevée.

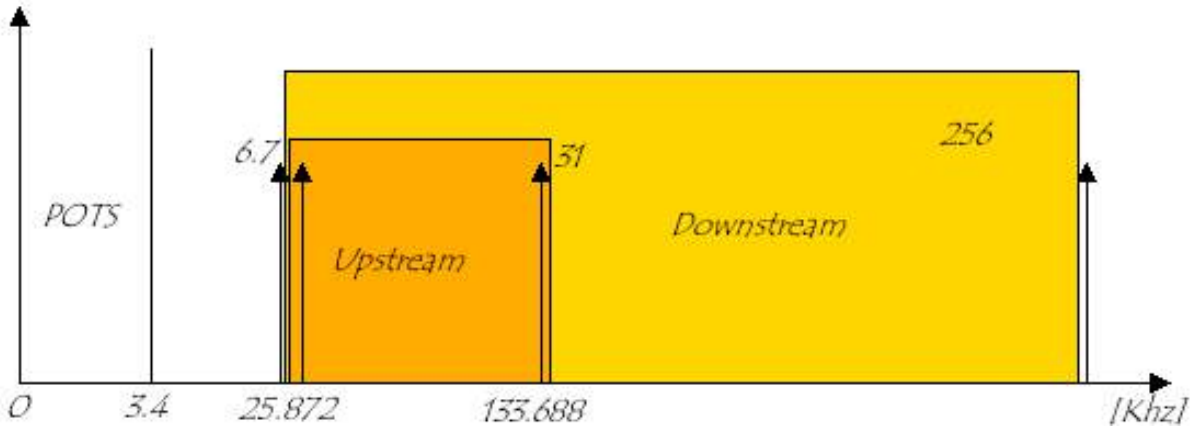


Fig. 25. Répartition du spectre de fréquences pour l'ADSL

Techniquement, au niveau de la centrale, un splitter "mélange" les signaux téléphoniques classiques véhiculant la voix avec les signaux à hautes fréquences de l'ADSL. Ce signal complexe est envoyé sur la ligne et, à la réception, un splitter ou filtre sépare les deux groupes de signaux en vue d'envoyer uniquement les hautes fréquences vers le modem ADSL de l'utilisateur. Ce modem intelligent décode les signaux et supervise les transmissions.

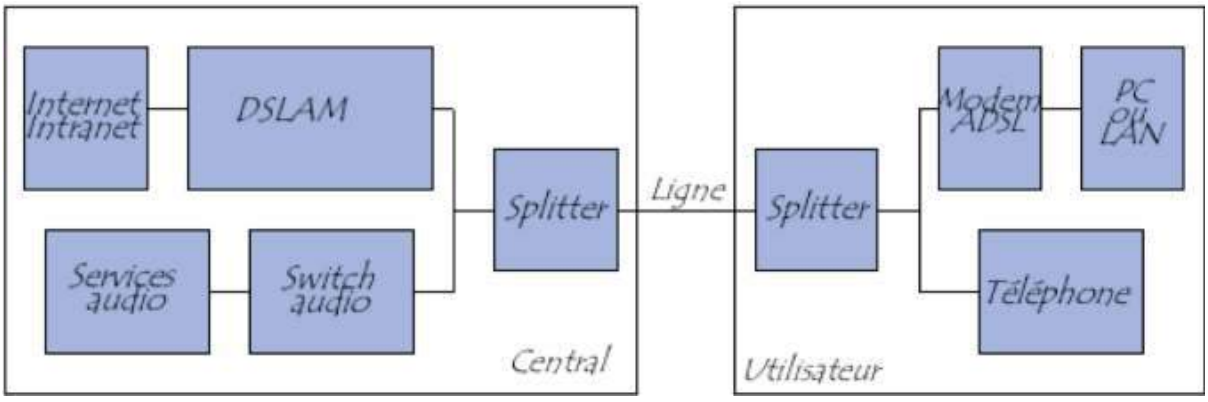


Fig. 26. Schéma de principe d'une liaison ADSL

[5] A noter que le canal 32 est réservé, que les canaux 16 et 64 sont utilisés pour transporter un signal pilote et que les canaux 250 à 256 ne sont utilisables que sur des lignes de raccordement de faible longueur car, au dessus de 1 MHz, les perturbations sont trop grandes pour permettre un flux stable à longue distance.

La vitesse effective de transmission est fixée de manière automatique et dynamique par le modem qui recherche le canal qui donnera la vitesse maximale possible en fonction de la qualité de la ligne de raccordement. En pratique, le modem entame la transaction sur le canal 32 et essaye successivement tous les autres canaux en vue de rechercher le meilleur (non encore occupé!). Les débits ascendants peuvent aller de 128kbps à 1Mbps et des débits descendants de 600kbps à 7Mbps, pour une longueur maximale de boucle locale de 5,4 km.

L'ADSL est actuellement une des seules technologies disponibles sur le marché qui offrent le transport de la TV/vidéo sous forme numérique (MPEG1 ou MPEG2) en utilisant un raccordement téléphonique.

6.- LES ECRANS D'AFFICHAGE

6.1.- Le tube cathodique

Le tube cathodique, développé depuis les débuts des années 50, permet d'afficher une image couleur produite par un flux d'électrons qui frappe la surface active du tube.

Fig. 27. Un écran à tube cathodique

Chaque point lumineux (pixel) d'un écran couleur est constitué de trois matières, autrefois trois disques disposés en triangle équilatéral, aujourd'hui trois rectangles juxtaposés horizontalement. La face du tube est donc recouverte de triples points minuscules. Chacune de ces matières produit une couleur si elle est soumise à un flux d'électrons. Les couleurs sont les suivantes : le rouge, le vert et le bleu. Il y a trois canons à électrons, un par couleur, et chaque canon ne peut allumer que les points d'une couleur. Un masque (plaque métallique percée de trous : un par pixel) est disposé dans le tube juste avant la face, pour éviter qu'un canon ne déborde sur l'autre.

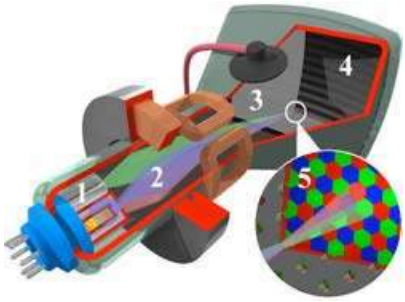


Fig. 28. Principe de l'écran couleur à trois canons

Les tubes cathodiques ont une intensité lumineuse qui n'est pas linéaire en fonction de l'intensité du flux d'électrons et qui doit être corrigée pour permettre de voir l'image sous ses vraies couleurs, ce qui est très important dans l'imprimerie entre autres (on appelle cela le gamma).

6.2.- Les écrans à cristaux liquides (LCD)

Un écran LCD utilise la polarisation de la lumière pour produire une image par transparence. Il se compose de plusieurs couches superposées dont deux polariseurs prenant en sandwich une couche de cristaux liquides en phase nématique dont on peut faire varier localement l'orientation en fonction d'un champ électrique. En fonction de la polarisation, la lumière passe ou ne passe pas. Du point de vue optique, l'écran à cristaux liquides est un dispositif passif (il n'émet pas de lumière) dont la transparence varie; il doit fonctionner par réflexion ou être rétro-éclairé.

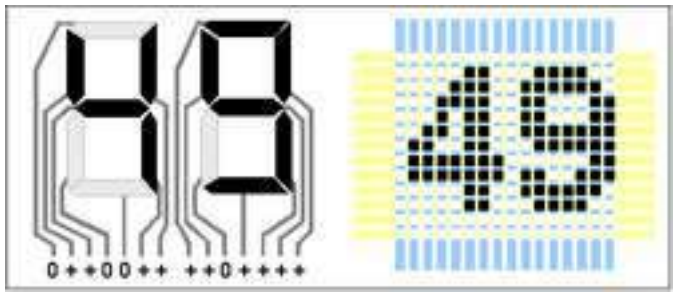


Fig. 29. Affichage LCD simple par segments et par pixels

http://fr.wikipedia.org/wiki/Image:LCD_layers.png

1. Filtre vertical qui polarise la lumière entrante.
2. Substrat de verre sur lequel une matrice transparente d'électrodes est déposée (sous forme d'un film métallique mince).
3. Cristaux liquides " nématiques" qui, en fonction du champ électrique induit entre les électrodes, fait "tourner" le plan de polarisation de la lumière.
4. Substrat de verre formant la seconde électrode.
5. Second filtre polarisant croisé avec le premier
6. Surface qui renvoie l'image ou une source uniforme d'éclairage

Pour les écrans couleur, on ajoute un masque coloré.

Fig. 30. Structure d'un écran LCD

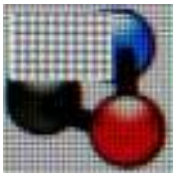


Fig.31. Vue agrandie des pixels d'un écran LCD couleur laissant apparaître le masque coloré

Il existe plusieurs modes d'éclairage adapté à chaque contexte d'utilisation qui doit tenir compte de la relative transparence des dispositifs à cristaux liquides : 15% pour les afficheurs monochromes et moins de 5% pour les écrans couleur du fait de l'interposition du masque coloré.

Éclairage réflectif

L'écran fonctionne seulement par réflexion de la lumière incidente. L'avantage est une faible consommation électrique et une luminosité naturellement adaptée à l'éclairage ambiant mais ils sont illisibles quand l'éclairage ambiant est faible.

Éclairage transmissif

L'écran fonctionne avec un rétro-éclairage (TV, moniteur informatique, appareil photo et caméra) fourni par une ou des lampes à décharge à cathode froide dont la lumière est diffusée par des réseaux de prismes orthogonaux et qui éclaire la surface d'une façon homogène. Ils fonctionnent très bien lorsque la luminosité ambiante est limitée (leur utilisation reste problématique en plein soleil!) mais ils consomment une énergie importante pour l'éclairage et la durée de vie des lampes limite leur durée de fonctionnement[6].

Fig. 32. Un écran LCD de grande taille

[6] Des prototypes de LCD rétro-éclairés par une matrice de diodes électroluminescentes (LED) blanches ont été présentés; ils améliorent nettement l'uniformité d'éclairage et promettent une durée de vie équivalente à celle du panneau LCD.

6.3.- Projecteur DATA

L'éclairage par transmission est également utilisé dans les projecteurs, où l'image d'un écran LCD couleur de petite taille (environ 2 cm de diagonale) est projetée par un dispositif optique comparable à un projecteur de diapositive utilisant une lampe halogène de forte puissance.

Les écrans modernes utilisent la technologie TFT, dite de matrice active, qui remplace la grille d'électrodes avant par une seule électrode et la grille arrière par une matrice de transistors en film mince (TFT- Thin-Film Transistor), un transistor par pixel (trois par pixel en couleur). Cette technique permet de mieux contrôler le maintien de tension de chaque pixel pour améliorer le temps de réponse et la stabilité de l'affichage. Les écrans TFT hors tension sont noirs.

Le processus de fabrication des dalles LCD est très complexe et utilise une succession de machines de très haute précision en atmosphère contrôlée. La vitre avant reçoit successivement les pigments du masque coloré. Les différentes lames qui formeront l'écran sont assemblées par collage (qui doit être extrêmement précis - de l'ordre du micromètre - pour assurer une parfaite correspondance entre les couches).

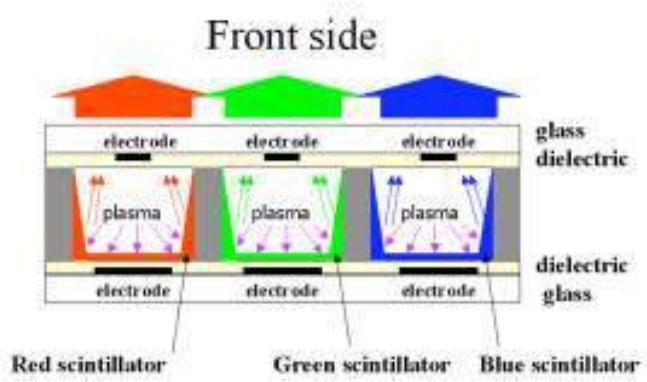
On injecte ensuite les cristaux liquides entre deux couches dans un espace de 10 à 20 μm d'épaisseur, soit moins de 1/100 de l'épaisseur totale (ce qui rend très long le remplissage des écrans de grande taille).

6.4.- Ecran Plasma

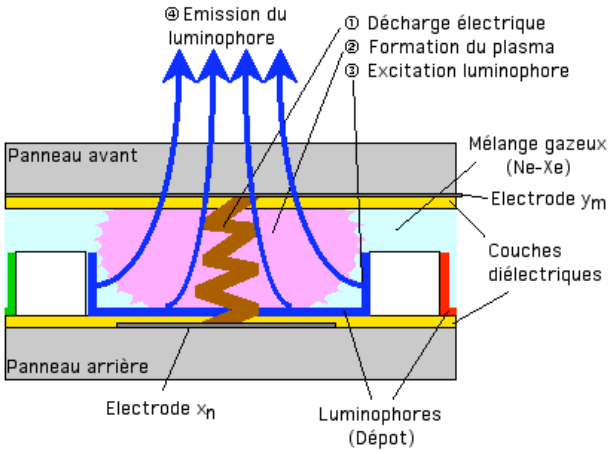
Le nom technique de l'écran plasma est le PDP (Plasma Display Panel). Il se compose d'un "sandwich" formé de deux feuilles de matériau transparent, entre lesquelles est incorporée une structure en nid d'abeille composée de nombreuses cellules. Chaque cellule d'un écran à plasma est une source de lumière indépendante, de surface inférieure au millimètre carré, et dont le fonctionnement est très voisin de celui d'une lampe fluorescente (lampe à décharge électrique). Un pixel est composé de trois cellules de décharge émettant de la lumière dans les trois couleurs fondamentales. Le plasma (milieu ionisé) de chaque cellule est créé par le passage du courant dans un gaz rare (mélange xénon-néon) et émet des photons ultra-violets convertis en photons visibles par des luminophores de différentes couleurs disposés sur les parois de cette cellule (fig. 33 & 34).

La difficulté est de contrôler électroniquement chaque micro-plasma à l'aide d'électrodes lignes et colonnes disposées sur deux dalles de verre en vis à vis et séparées d'environ 100 microns. En fait, le plasma est généré de façon impulsionnelle pendant des temps très courts (environ 100 nanosecondes). Pour moduler l'intensité de la lumière perçue par l'œil, on joue sur le nombre d'allumages de chaque micro-plasma pendant une image vidéo. On arrive ainsi à générer 256 niveaux d'intensité (" niveaux de gris ") pour chaque cellule de dé-

charge. Puisque l'on a trois cellules par pixel, on peut générer 256 x 256 x 256 soit plus de 16 millions de couleurs ! Sachant que l'on doit contrôler ainsi environ un million de cellules, on comprend la complexité de l'électronique de commande et le coût qui en résulte.



un pixel de 3 cellules



une cellule

http://pcml.univ-lyon1.fr/activite_scientifique/Luminophores/ecrplasma.html

Fig. 33. Principe de fonctionnement d'un pixel d'écran plasma

Fig. 34. Cellules excitées par décharge électrique

6.5.- Ecran électroluminescent organique (OLED Organic light-emitting diode)

L'écran est formé d'une myriade de diodes émettrices de lumière (LED) groupées par 3 (une rouge, une verte et une bleue) et formant les pixels d'un écran OLED. L'ensemble repose sur un « substrat » transparent, en verre ou en plastique souple.

Fig. 35. Un écran souple OLED

La technologie OLED possède de nombreux avantages par rapport aux LCD (faible consommation électrique, meilleur rendu des couleurs, meilleur contraste, minceur et souplesse du support, etc...), le seul point faible étant la durée de vie qui n'est pas encore optimale (aux alentours de 10 000 heures).

Cette technologie a vocation à remplacer peu à peu les affichages à cristaux liquides, d'abord dans les applications de petites dimensions telles que téléphones mobiles, écrans d'appareils numériques et ensuite, quand la fabrication sera mieux maîtrisée, dans les écrans de grande taille.

Table des matières

Le serveur Web concernant le cours

Introduction à l'informatique

H.P. Garnir

(2ème bac. biologie et géologie, et 1ère lic. biochimie)

http://www.ulg.ac.be/ipne/info

Ce serveur est local et son accès est limité par un mot de passe :

Le user name est : "info"
Le mot de passe est : "ok".

H.P. Garnir